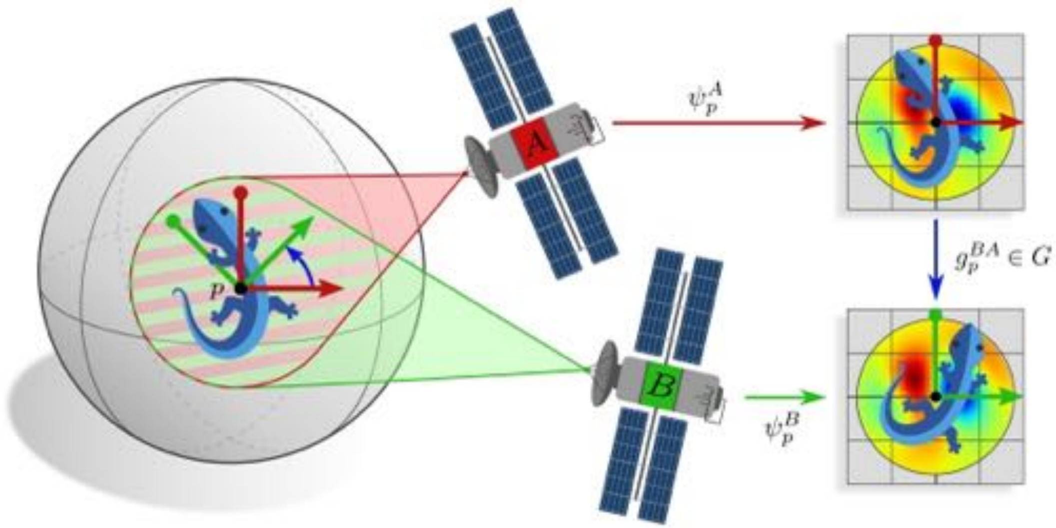
Microsoft

AI4Science

Max Welling

Overview

- Intro to Fifth Paradigm
- PDE surrogates
- Molecules

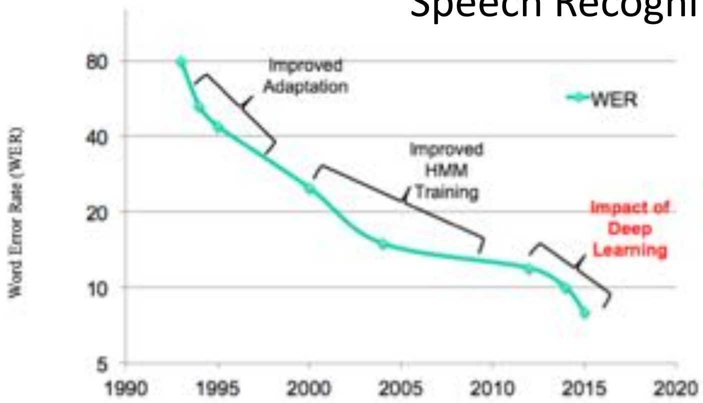


Picture created by Maurice Weiler

(Lizards adapted under the Creative Commons Attribution 4.0 International license by courtesy of Twitter.)

Deep Learning has been transforming AI

Speech Recognition



Source: Kartik Audhkhasi blog; https://minghsiehece.usc.edu/2017/04/the-machines-are-coming/

Natural Language Models

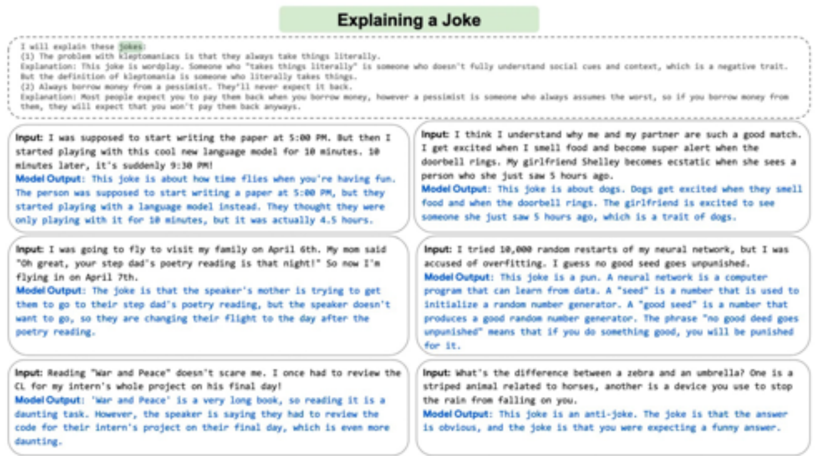
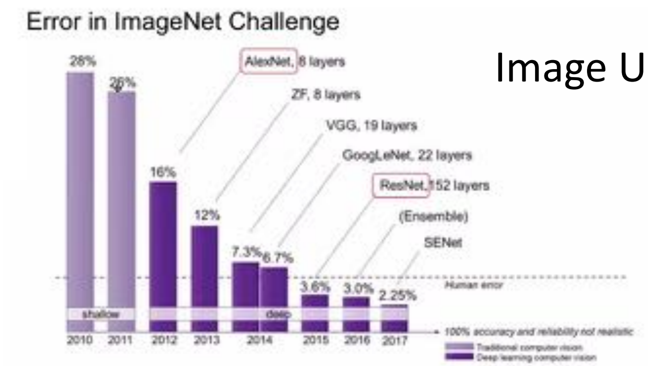


Image Understanding



(Source: [Angshuman Gosh](#) | DLDC 2021)

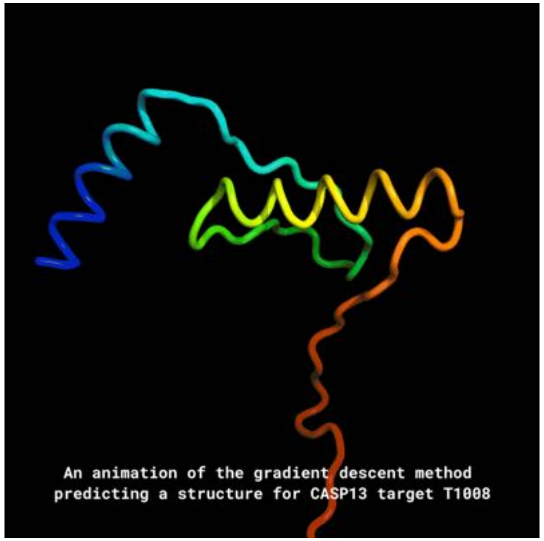
Text to Image Generative Models

A small cactus wearing a straw hat and neon sunglasses in the Sahara desert.

[Imagen Video (research.google)](#)

Deep learning will be transforming the natural sciences

Protein Folding

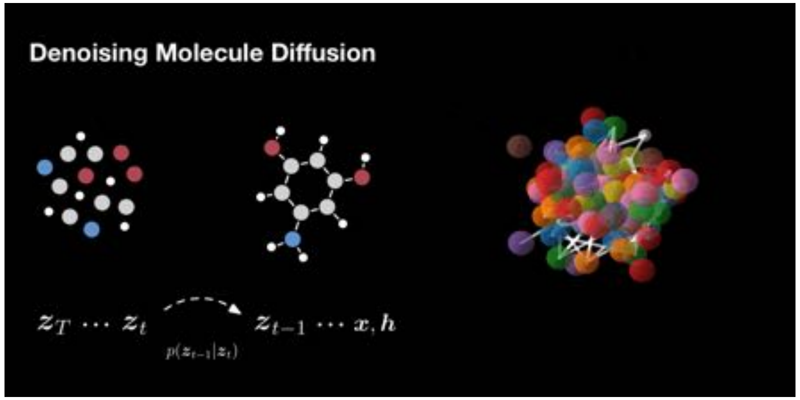


Highly accurate protein structure prediction with AlphaFold

John Jumper, Richard Evans, ... Demis Hassabis + Show authors

Nature **596**, 583–589 (2021) | Cite this article

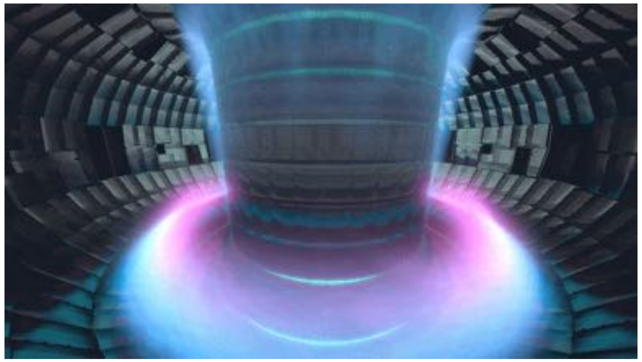
Molecule Generation



Equivariant Diffusion for Molecule Generation in 3D

Emiel Hoogeboom * [1] Victor Garcia Satorras * [1] Clément Vignac * [2] Max Welling [1]

Plasma Control



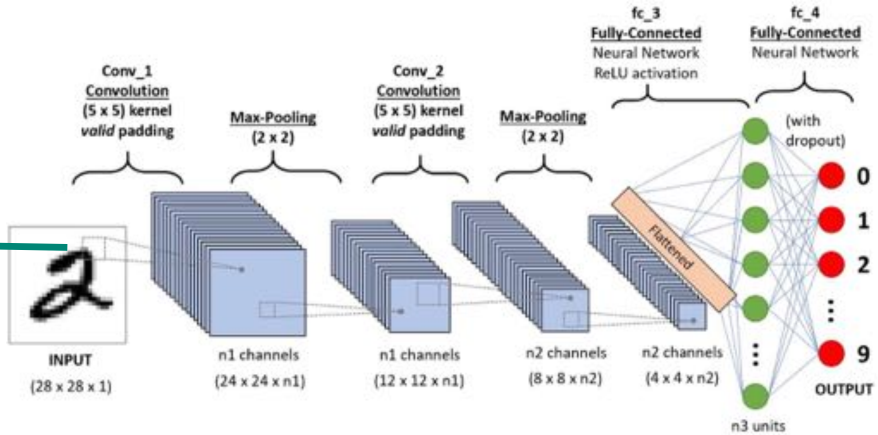
Magnetic control of tokamak plasmas through deep reinforcement learning

Jonas Degrave, Federico Felici, ... Martin Riedmiller + Show authors

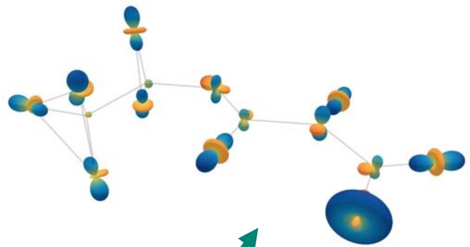
Nature **602**, 414–419 (2022) | Cite this article

The main tool: equivariant GNNs

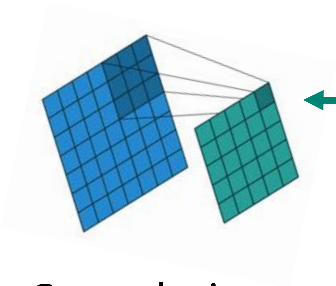
Convolutional Neural Network



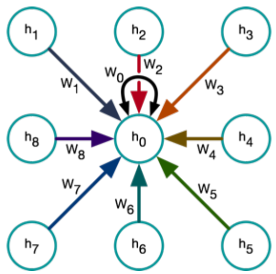
Equivariant Graph Convolution



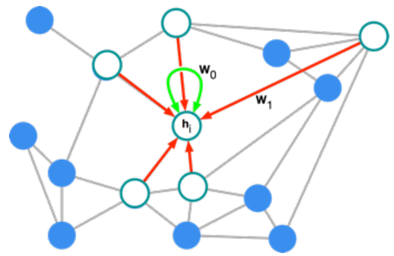
Convolution



Convolution as message passing

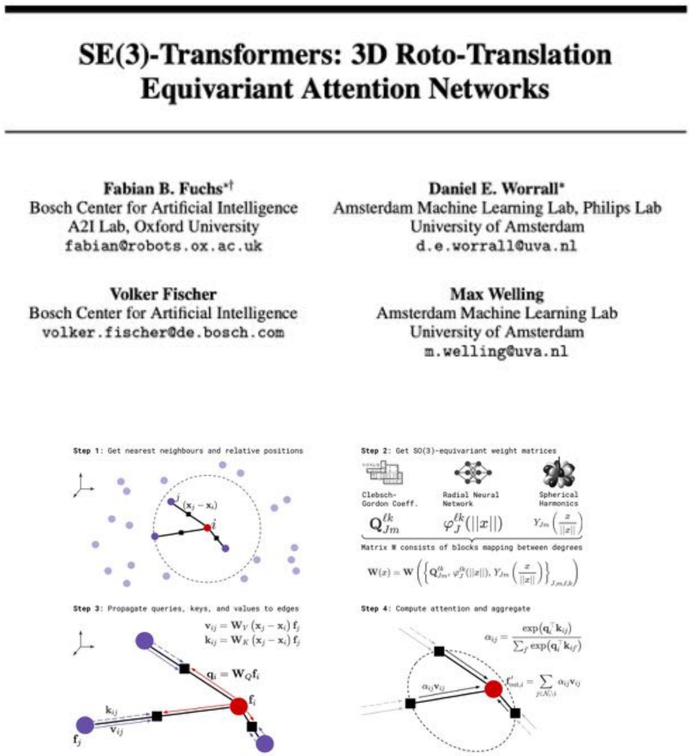


Graph Convolution as message passing



Further Reading

Generalized SE(3) Equivariant GNNs using higher order irreducible representations.



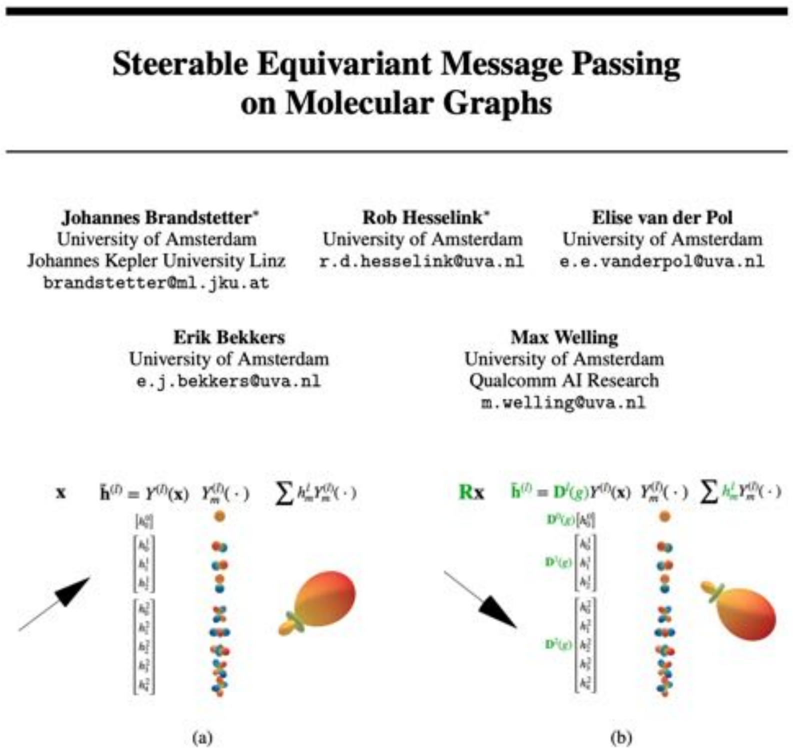
SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks

Fabian B. Fuchs*†
Bosch Center for Artificial Intelligence
A2I Lab, Oxford University
fabian@robots.ox.ac.uk

Daniel E. Worrall*
Amsterdam Machine Learning Lab, Philips Lab
University of Amsterdam
d.e.worrall@uva.nl

Volker Fischer
Bosch Center for Artificial Intelligence
volker.fischer@de.bosch.com

Max Welling
Amsterdam Machine Learning Lab
University of Amsterdam
m.welling@uva.nl



Steerable Equivariant Message Passing on Molecular Graphs

Johannes Brandstetter*
University of Amsterdam
Johannes Kepler University Linz
brandstetter@ml.jku.at

Rob Hesselink*
University of Amsterdam
r.d.hesselink@uva.nl

Elise van der Pol
University of Amsterdam
e.e.vanderpol@uva.nl

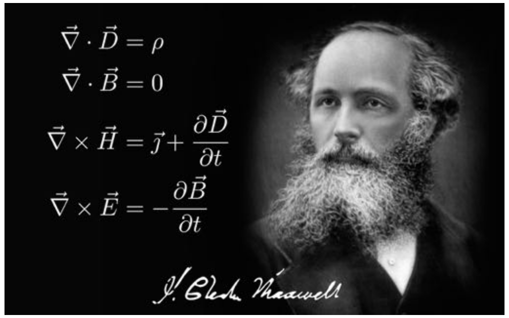
Erik Bekkers
University of Amsterdam
e.j.bekkers@uva.nl

Max Welling
University of Amsterdam
Qualcomm AI Research
m.welling@uva.nl

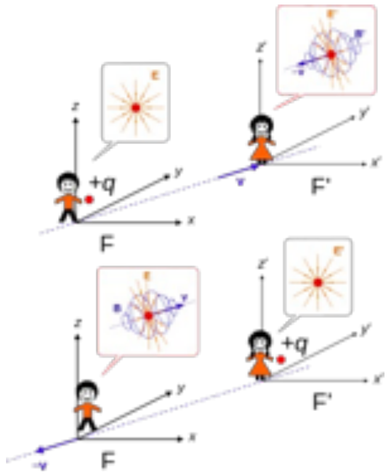
(a)

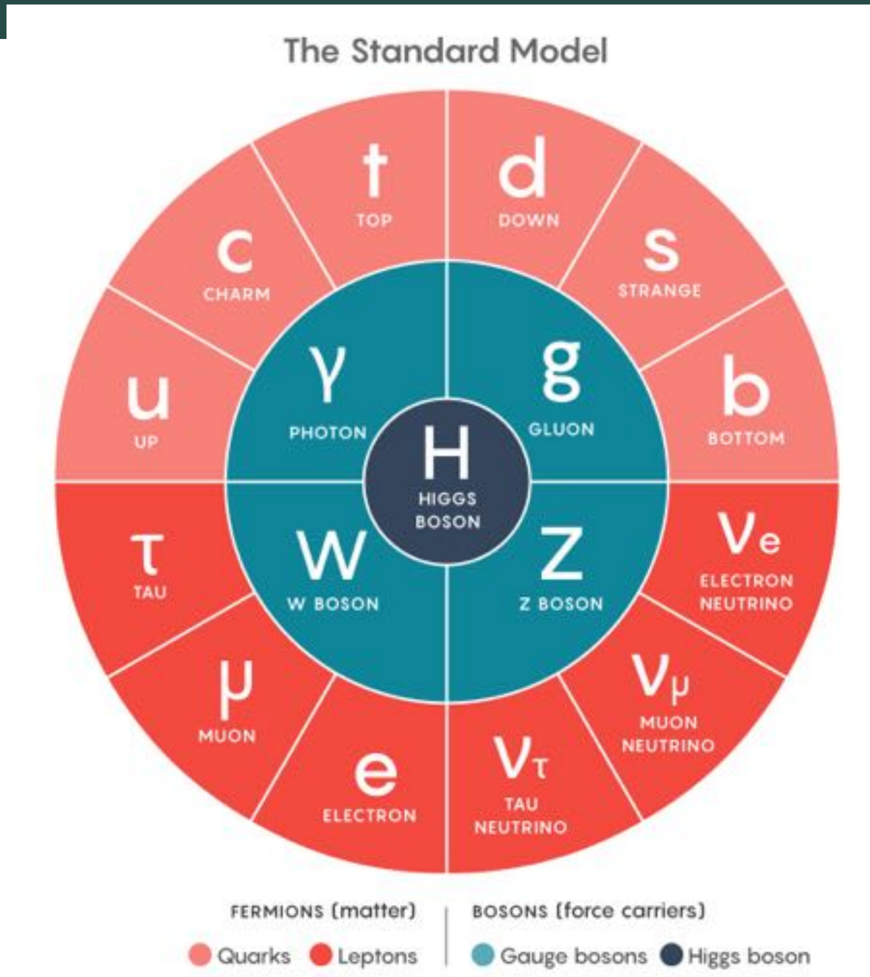
(b)

Symmetries & Equivariance

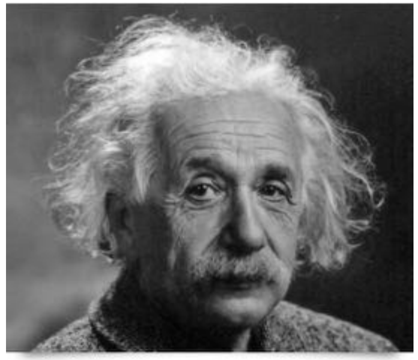


Electricity = Magnetism



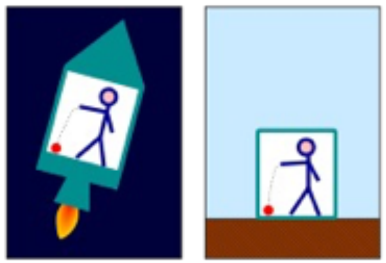


- ield.
-
-

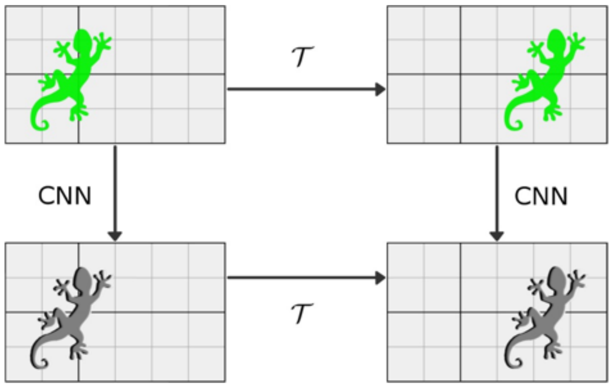


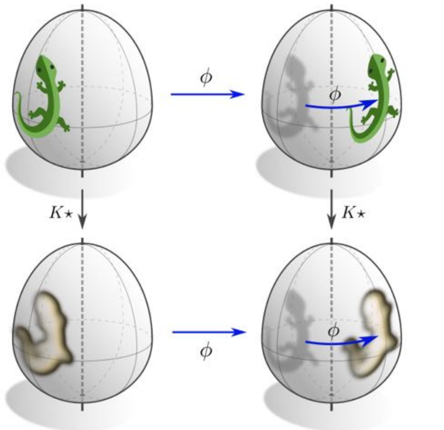
Gravity = Acceleration

$$R_{\mu\nu}-\frac{1}{2}R\,g_{\mu\nu}+\Lambda\,g_{\mu\nu}=\frac{8\pi G}{c^4}T_{\mu\nu}$$

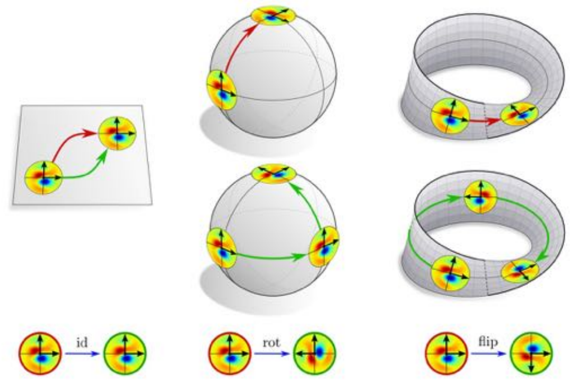


Equivariance





Equivariance on manifold



Gauge symmetries are needed to define proper convolutions on manifolds

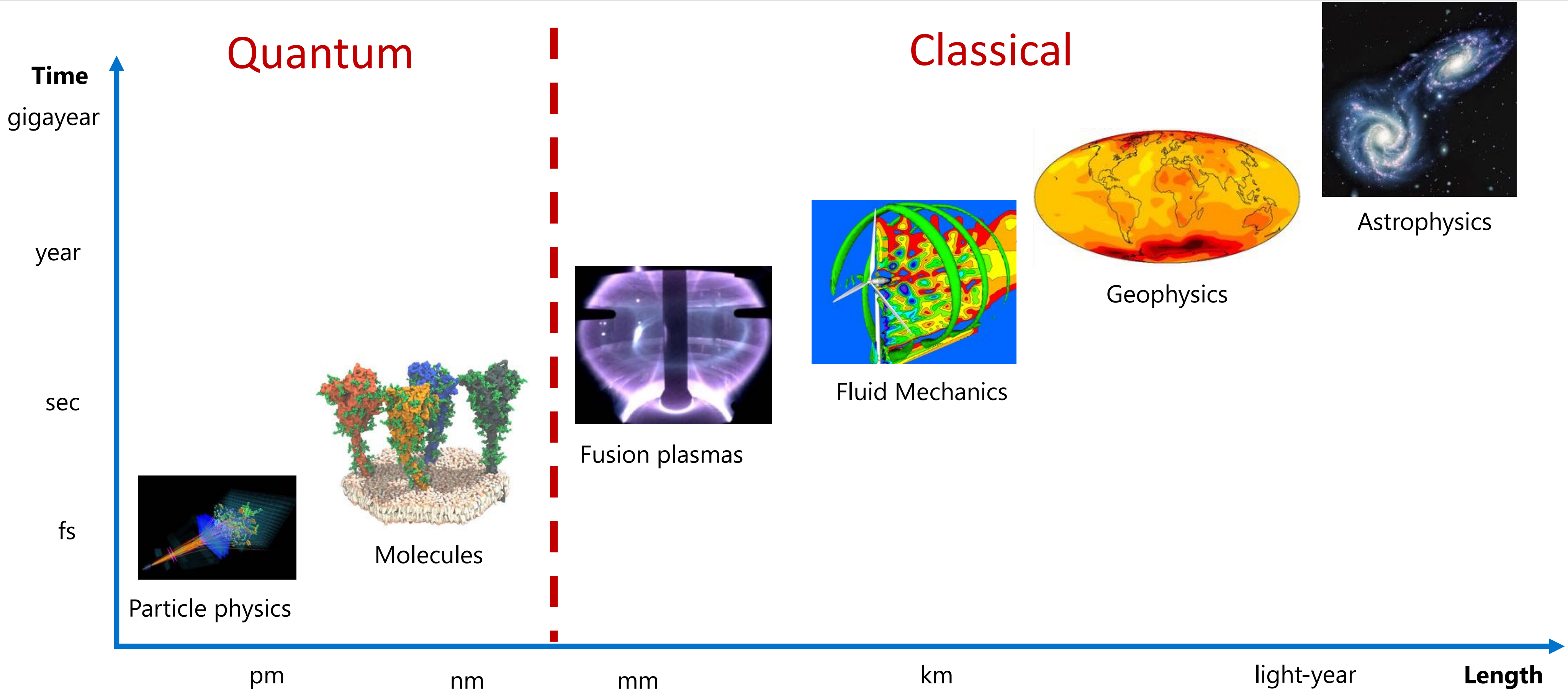
- Equivariance is good for:
  - Data efficiency
  - Disentangling pose and presence
  - Creates easy patterns for next layer

- First appearance in ML: Group CNNs
  Cohen & W. ’16, Dieleman et al, ‘16

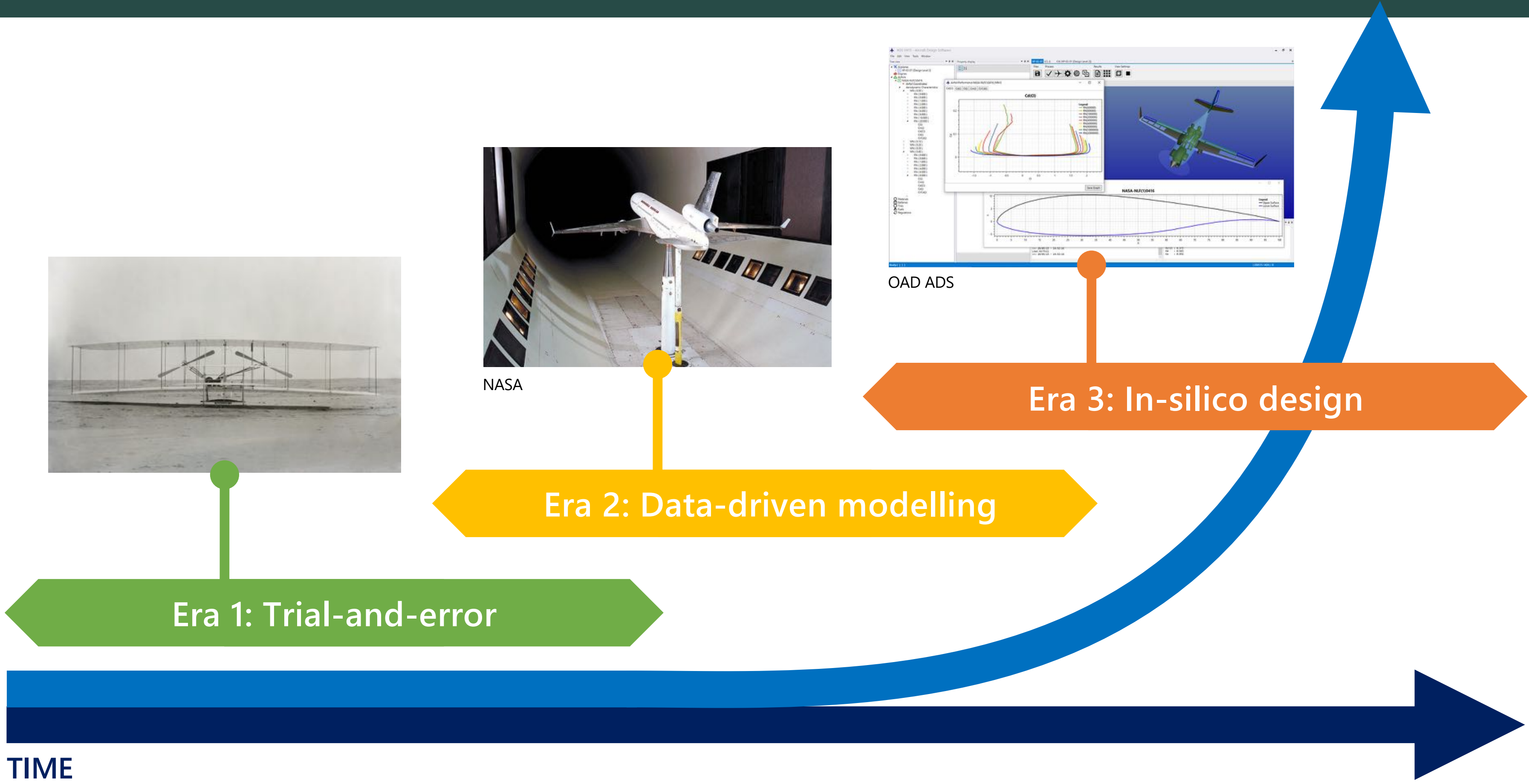
Picture created by Maurice Weiler

(Lizards adapted under the Creative Commons Attribution 4.0 International license by courtesy of Twitter.)

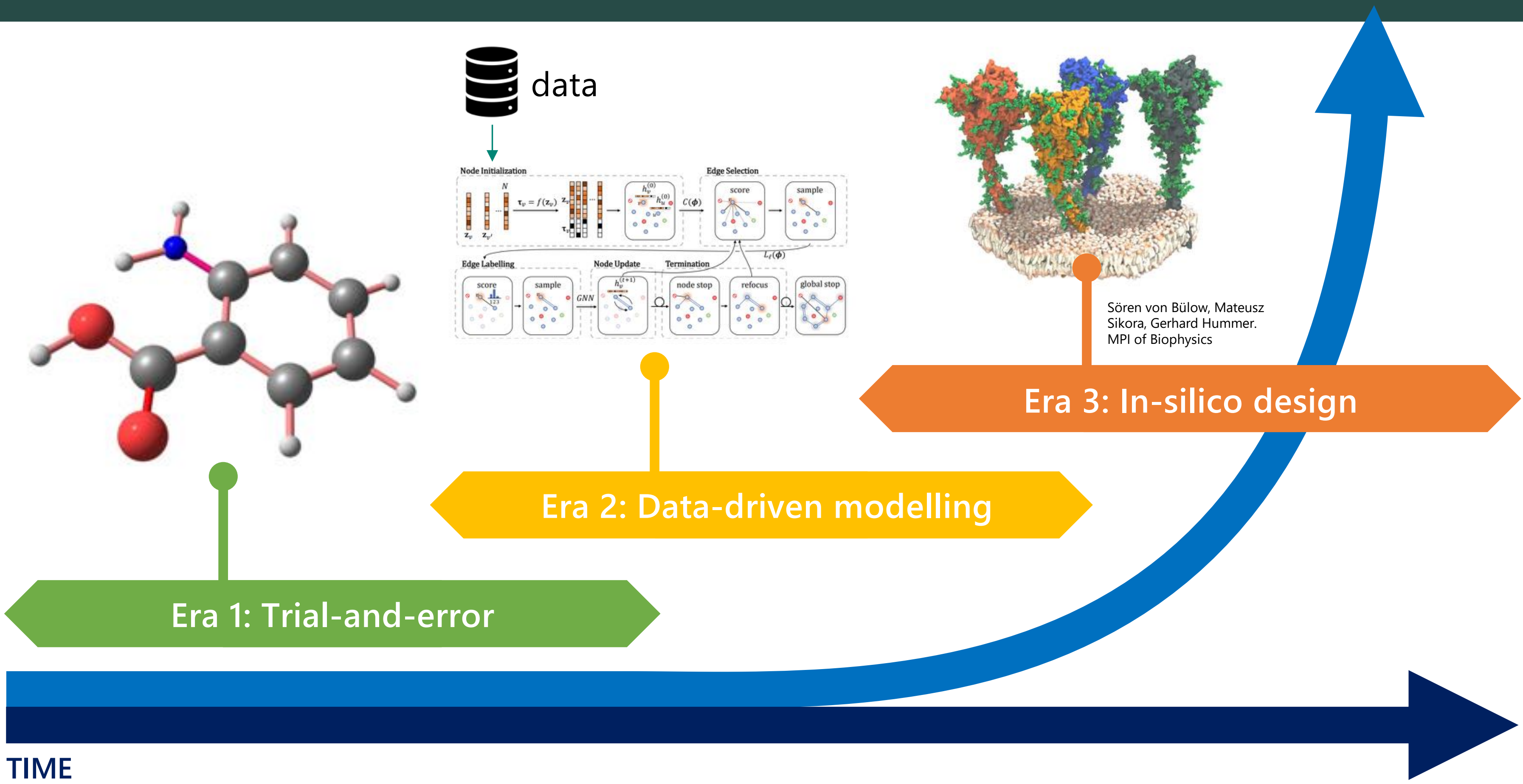
AI4Science: A Multi-Scale Scientific World



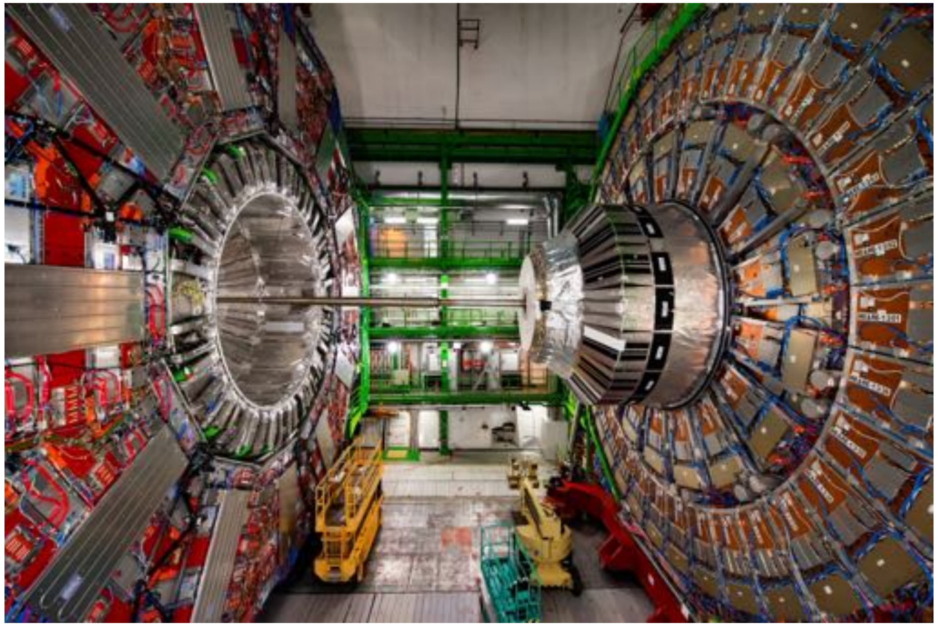
A New Paradigm of Scientific Discovery



A New Paradigm for Materials Design



Can we build a new kind of microscope?



LHC: The microscope of the particle physicists

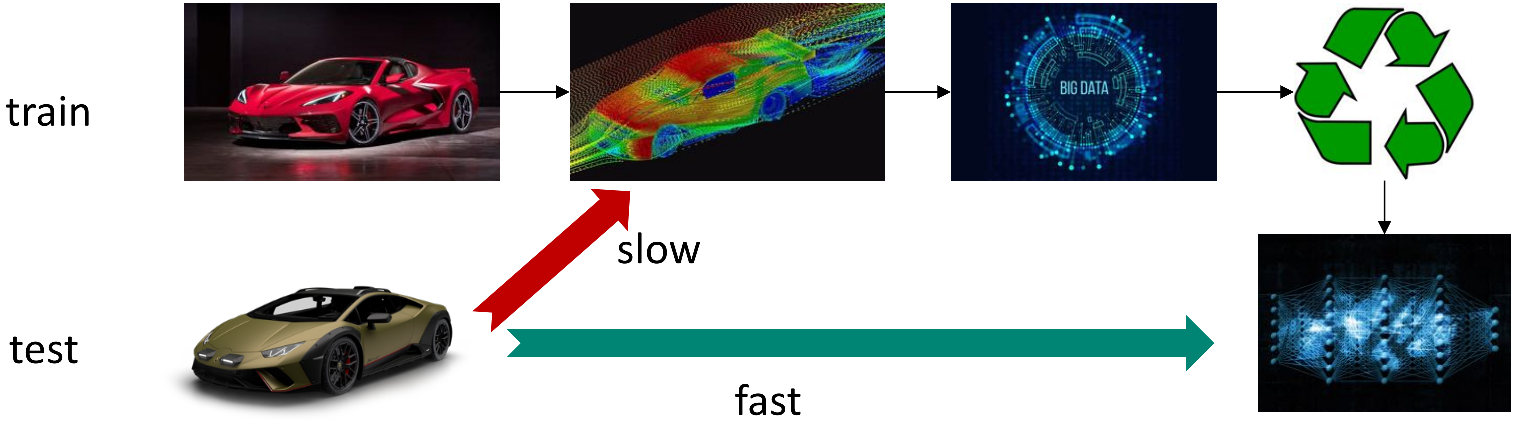
SKA: The telescope of the astronomers

The new microscope is computational

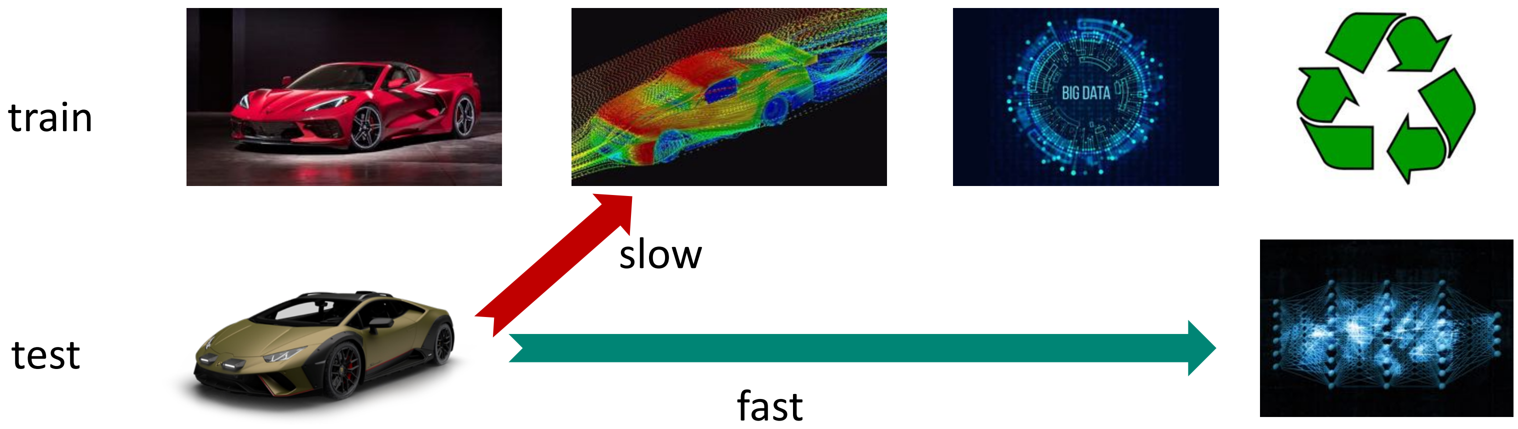
Large scale, self-learning simulations
on modern supercomputers

Amortization

- The usual paradigm is to ”solve” the physics equation through numerical methods
- Data is thrown then away!
- Fifth paradigm is recycling data and storing information in model parameters
- ML surrogate can shortcut expensive computation when pattern is seen before



Generalization

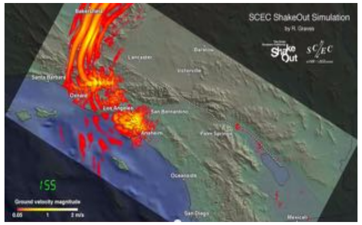


- When should we use the slow similar versus the fast emulator?
  - Simulator solves physics equations: generalizes well
  - Emulator is neural network model: may generalize poorly
- Know when you don’t know: *uncertainty quantification is key*

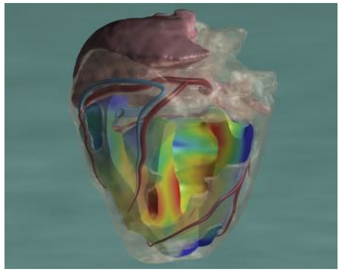
PDEs

Partial Differential Equations

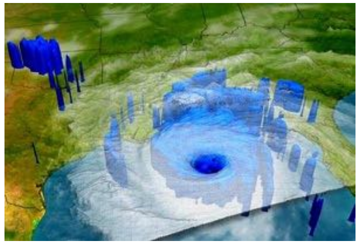
- PDEs are used throughout the sciences.
- We want to either replace or augment numerical schemes.



Earthquakes

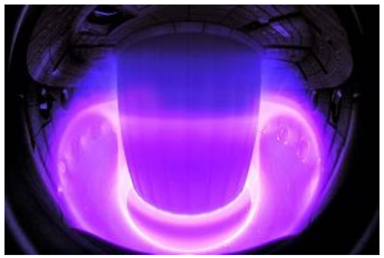


Heart dynamics

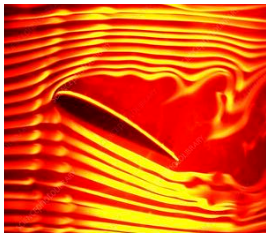


Weather prediction

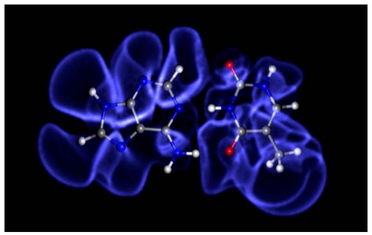
Galaxy collisions



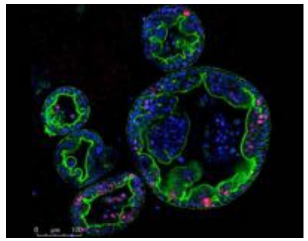
Plasma physics



Airplane design



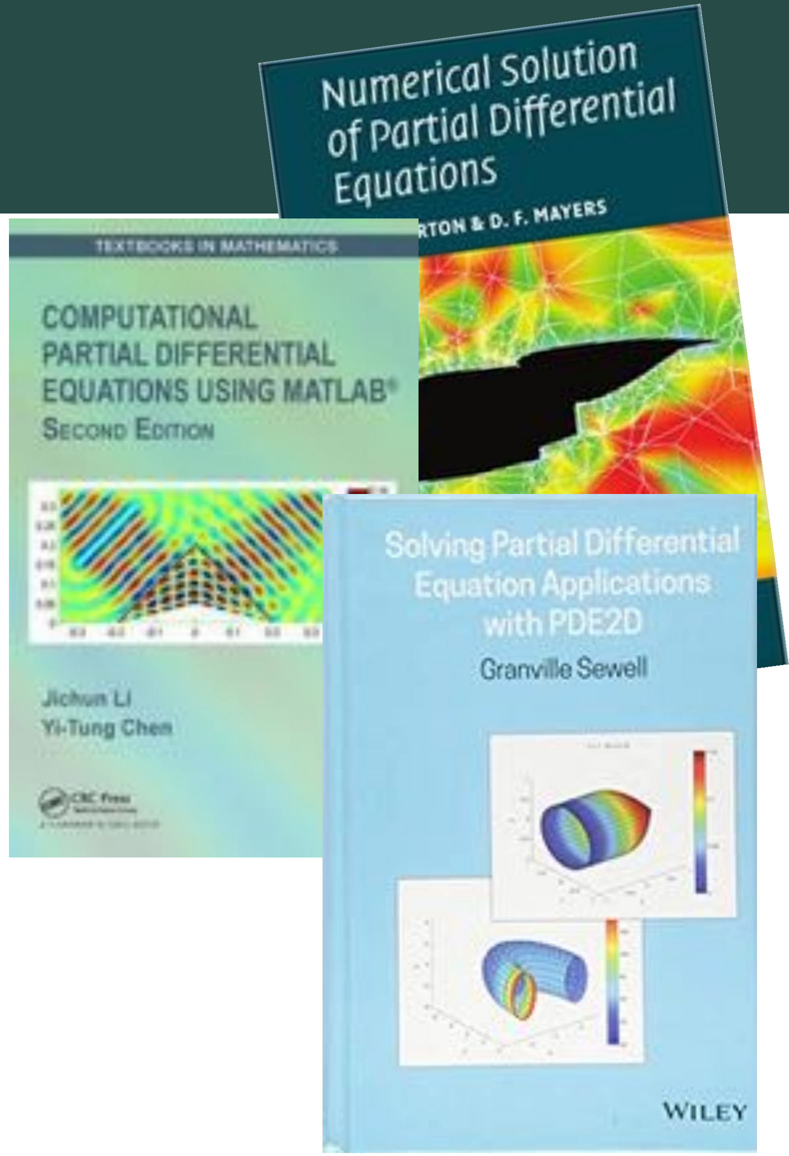
Electronic structure



Tumor growth

Numerical Solvers

- Requirements:
    - Accuracy
    - Stability over long rollouts
    - Speed
    - Computational cost
    - Easy to use
    - Uncertainty quantification
    - Generalize across:
        - Initial conditions
        - Boundary conditions
        - PDE parameters
        - Integration grid resolution
        - Integration grid regularity
        - Geometry
        - Topology
        - Dimensionality
    - …



PDEs

- Formulation of a (time-dependent) PDE:

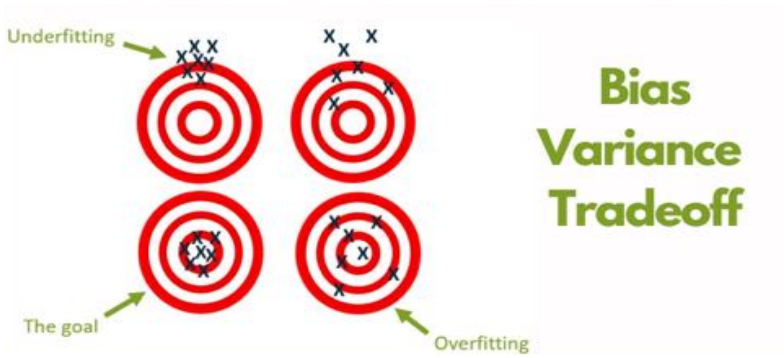
$$
\begin{aligned}
&\partial_t \mathbf{u} = F(t, \mathbf{x}, \mathbf{u}, \partial_{\mathbf{x}} \mathbf{u}, \partial_{\mathbf{xx}} \mathbf{u}, \ldots) && (t, \mathbf{x}) \in [0, T] \times \mathbb{X} \\
&\mathbf{u}(0, \mathbf{x}) = \mathbf{u}^0(\mathbf{x}), \qquad B[\mathbf{u}](t, x) = 0 && \mathbf{x} \in \mathbb{X},\ (t, \mathbf{x}) \in [0, T] \times \partial \mathbb{X}
\end{aligned}
$$

- Can ML be used to solve PDEs faster?
  - Think of solver as a differentiable iterative program: optimize its (hyper)parameters from data
  - Use either real data and/or simulated data to train ML models
  - Key question: how do ML PDE surrogates generalize across ICs, BCs, parameter perturbations, dimensions?

Data from numerical solver

Improve surrogate ML model to solve PDE faster next time

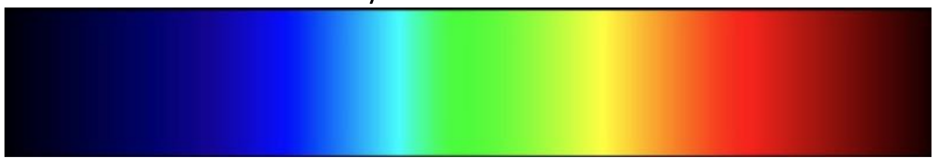
Generalization, Inductive Bias & Data



Best of both worlds?

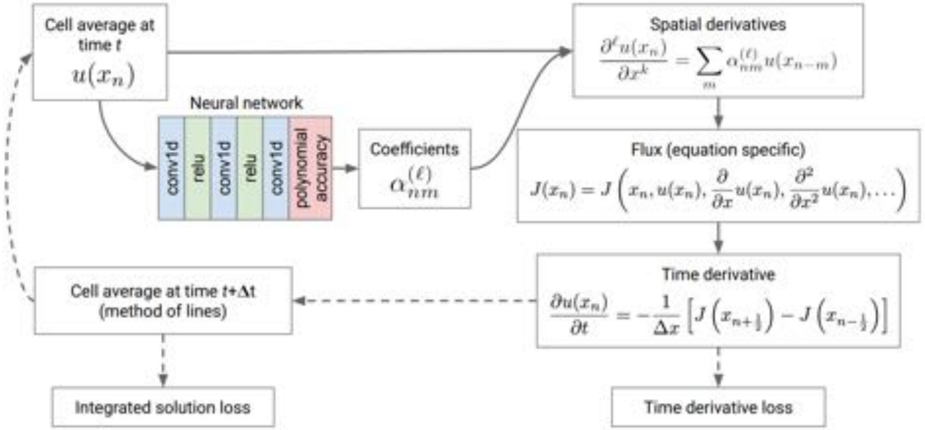
Slow,
Error guarantees,
Few parameters (e.g. RK)
Large inductive bias
No data
Generalization?

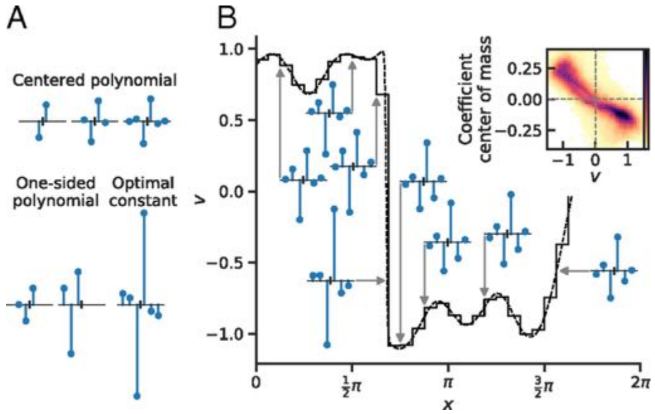
Numeric solver | hybrid solver | ML solver



Fast,
No error guarantees,
Lots of parameters (e.g. deep NN)
Little inductive bias
Lots of data
Generalization?

First attempts: Learning Stencils



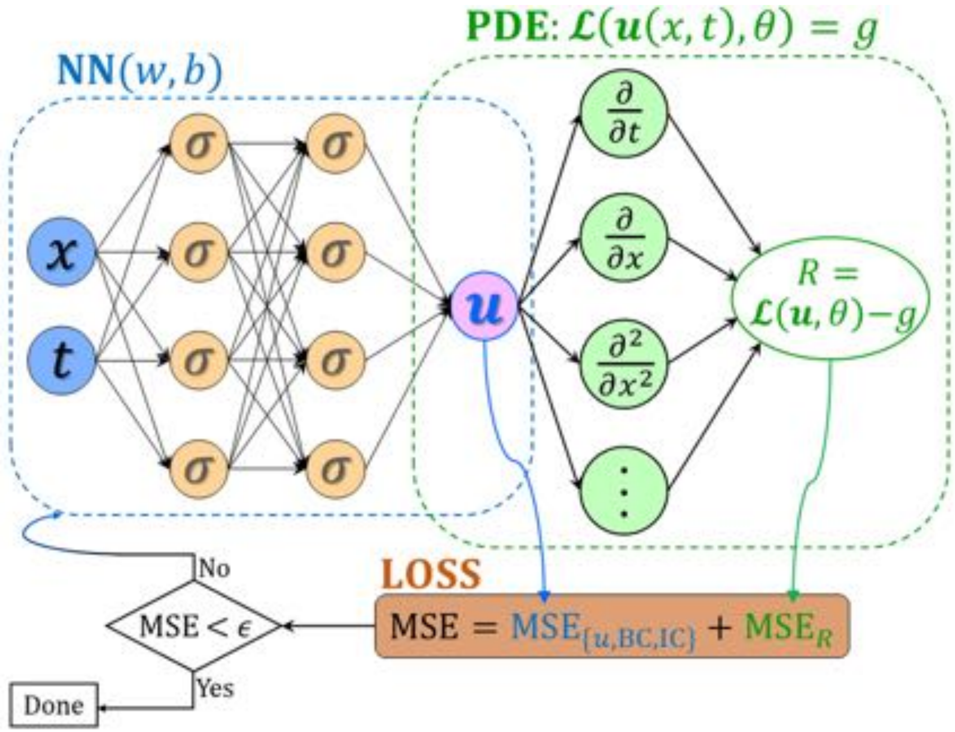


Yohai Bar-Sinai, Stephan Hoyer, Jason Hickey, and Michael P. Brenner. Learning data-driven discretizations for partial differential equations. Proceedings of the National Academy of Sciences, 116(31):15344–15349, Jul 2019. ISSN 1091-6490. doi: 10.1073/pnas.1814058116.

Solution approximators

PINNs:
Raissi et al.
Journal of Computational physics 2017

- PINN-like approaches (implicit function approximators):
- Inverse problems (learn PDE parameters)
- Good for high-dimensional problems

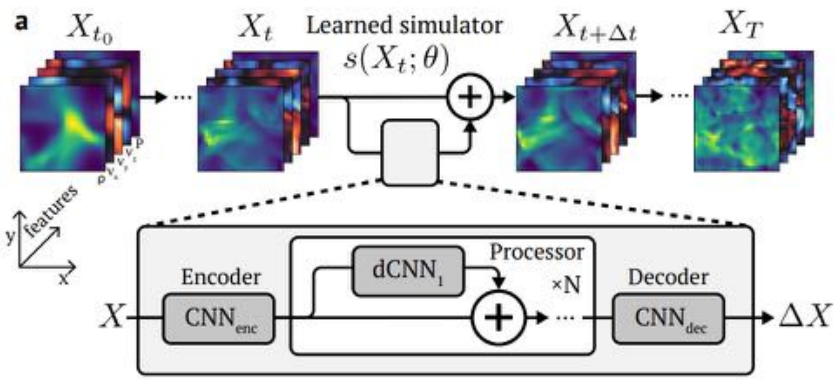


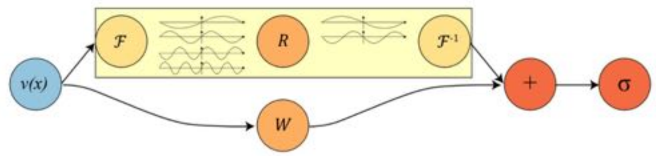
Neural operators

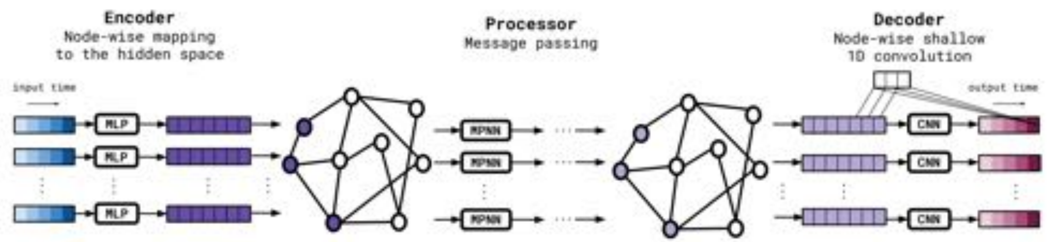
- Operator learning:
  - Map one solution to another solution
  - Method approximately independent from grid
  - Ideally generalizes to different grids, initial & boundary conditions, ...

DeepONet:
Lu et al.
Nature Machine Intelligence 2019

Fourier Neural Operator(FNO):
Li et al.
ICLR2021





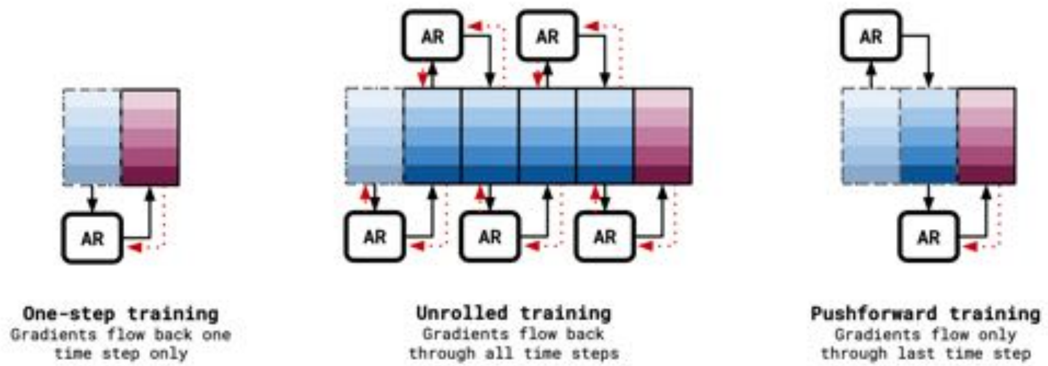


Training a Neural PDE solver

- Generate ”data” from classical solver.
- Train model by minimizing Loss function:

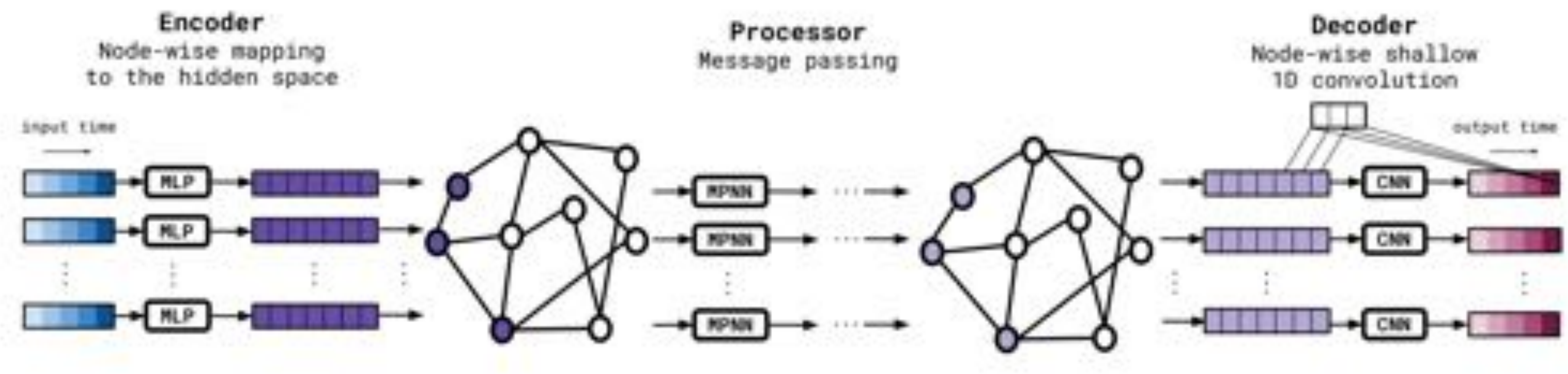
$$L_{\text{stability}} = \mathbb{E}_k \mathbb{E}_{\mathbf{u}^{k+1}|\mathbf{u}^k, \mathbf{u}^k \sim p_k} \left[\mathbb{E}_{\epsilon|\mathbf{u}^k} \left[\mathcal{L}(\mathcal{A}(\mathbf{u}^k + \epsilon), \mathbf{u}^{k+1}) \right] \right]$$

with $(\mathbf{u}^k + \epsilon) = \mathcal{A}(\mathbf{u}^{k-1})$



- We train to predict the right answer from a noisy input.
- Noise is given by numerical integration errors

Encode – Process - Decode



x_i location

u_i^k field variable at x_i at time k

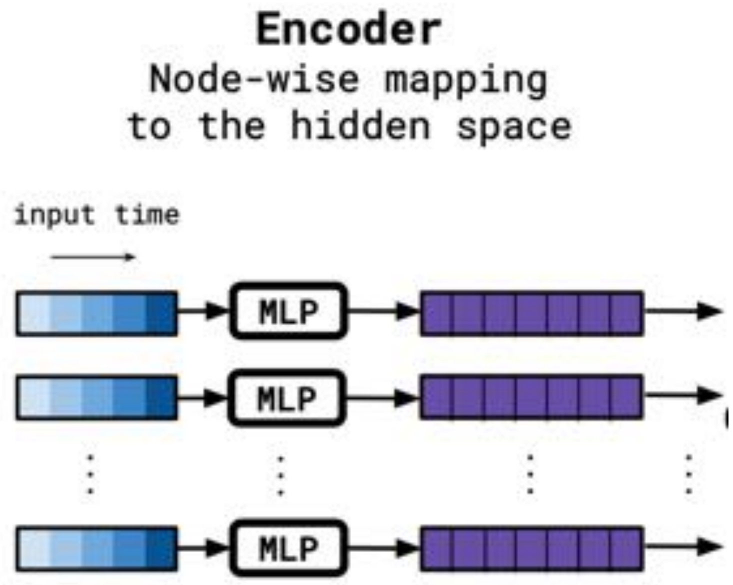
f_i^m GNN feature at x_i at layer m

θ_{PDE} other properties such as boundary conditions, PDE parameters etc.

Encoder

- Embed node information on graph:

$$\mathbf{f}_i^0 = \epsilon^v([\mathbf{u}_i^{k-K:k}, \mathbf{x}_i, t_k, \boldsymbol{\theta}_{\text{PDE}}])$$



Processor: GNN Message Passing on Irregular Grid

- Create irregular integration grid with the following information on nodes:

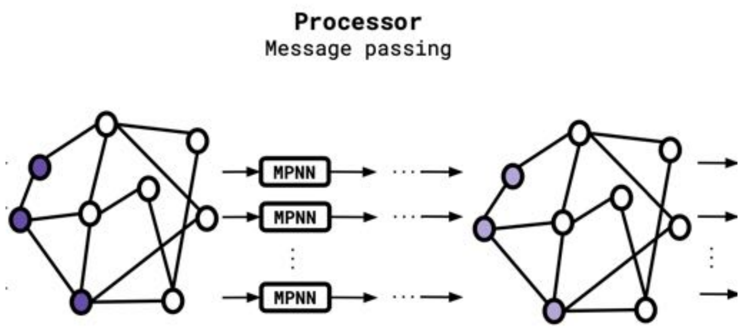
x_i location

u_i^k field variable at x_i at time k

f_i^m GNN feature at x_i at layer m

θ_{PDE} other properties such as boundary conditions, PDE parameters etc.

- Use GNN to process information:



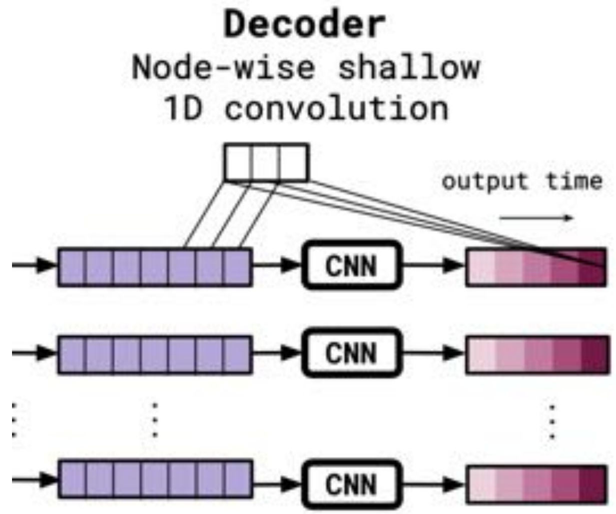
$$\text{edge } j \to i \text{ message:} \qquad \mathbf{m}_{ij}^m = \phi\left(\mathbf{f}_i^m, \mathbf{f}_j^m, \mathbf{u}_i^{k-K:k} - \mathbf{u}_j^{k-K:k}, \mathbf{x}_i - \mathbf{x}_j, \boldsymbol{\theta}_{\mathrm{PDE}}\right)$$

$$\text{node } i \text{ update:} \qquad \mathbf{f}_i^{m+1} = \psi\left(\mathbf{f}_i^m, \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{ij}^m, \ \boldsymbol{\theta}_{\mathrm{PDE}}\right),$$

Decode

$$d_i^1, d_i^2, ..., d_i^M = CNN(f_i^1, f_i^2, ..., f_i^M)$$

$$\mathbf{u}_i^{k+\ell} = \mathbf{u}_i^k + (t_{k+\ell} - t_k)\mathbf{d}_i^\ell$$



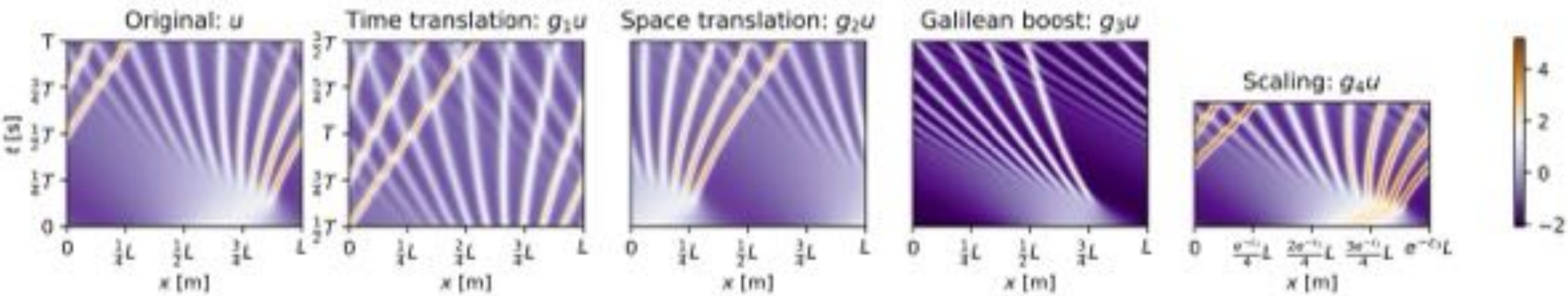
Handling Data Sparsity Symmetries: Korteweg-de Vries Eqn.

$$\mathbf{\Delta}((x,t),u^{(3)}) = u_t + uu_x + u_{xxx} = 0$$

Periodic BCs

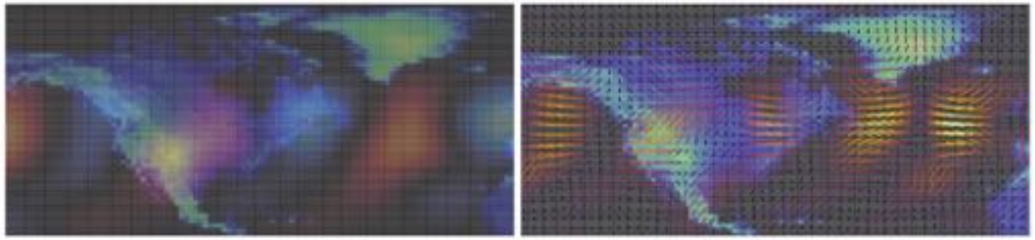
$$g = g_1(\epsilon_1)g_2(\epsilon_2)\cdots g_d(\epsilon_d)$$

$$\begin{aligned}
g_1(\epsilon)(x,t,u) &= (x, t+\epsilon, u) && \text{time shift,}\\
g_2(\epsilon)(x,t,u) &= (x+\epsilon, t, u) && \text{space shift,}\\
g_3(\epsilon)(x,t,u) &= (x+\epsilon t, t, u+\epsilon) && \text{Galilean boost,}\\
g_4(\epsilon)(x,t,u) &= (e^{\epsilon}x, e^{3\epsilon}t, e^{-2\epsilon}u) && \text{scaling,}
\end{aligned}$$



Brandstetter et al, 2022

PDEs can be solved many times faster with NNs



(a) Scalar pressure field **(b)** Vector wind velocity field

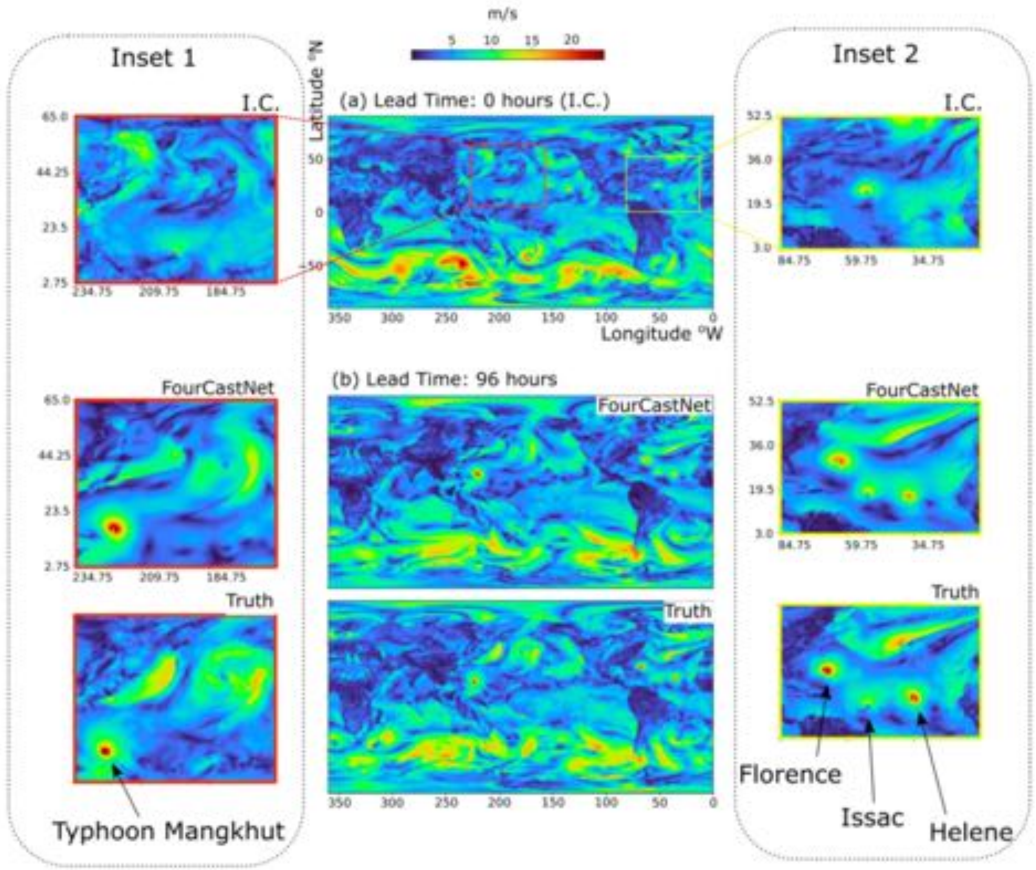
Clifford Neural Layers for PDE Modeling

Johannes Brandstetter[1], Rianne van den Berg[1], Max Welling[1], and Jayesh K. Gupta[2]
[1]Microsoft Research Amsterdam, [2]Microsoft Autonomous Systems and Robotics Research

FOURCASTNET: A GLOBAL DATA-DRIVEN HIGH-RESOLUTION WEATHER MODEL USING ADAPTIVE FOURIER NEURAL OPERATORS

A PREPRINT

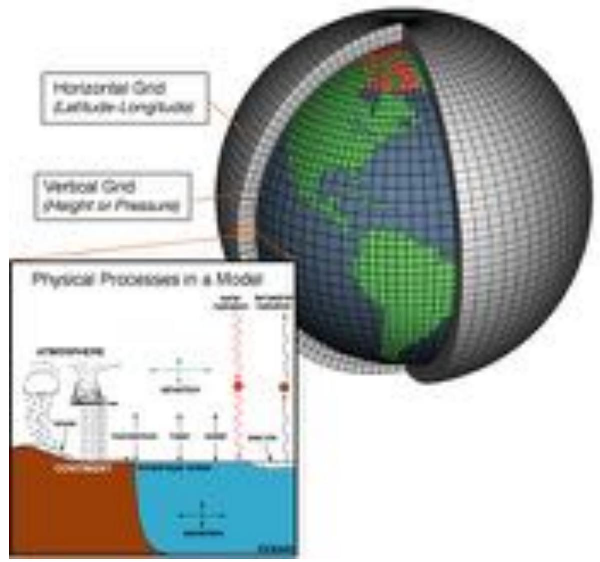
Jaideep Pathak NVIDIA Corporation, Santa Clara, CA 95051
Shashank Subramanian Lawrence Berkeley National Laboratory, Berkeley, CA 94720
Peter Harrington Lawrence Berkeley National Laboratory, Berkeley, CA 94720
Sanjeev Raja University of Michigan, Ann Arbor, MI 48109
Ashesh Chattopadhyay Rice University, Houston, TX 77005
Morteza Mardani NVIDIA Corporation, Santa Clara, CA 95051
Thorsten Kurth NVIDIA Corporation, Santa Clara, CA 95051
David Hall NVIDIA Corporation, Santa Clara, CA 95051
Zongyi Li California Institute of Technology, Pasadena, CA 91125; NVIDIA Corporation, Santa Clara, CA 95051
Kamyar Azizzadenesheli Purdue University, West Lafayette, IN 47907
Pedram Hassanzadeh Rice University, Houston, TX 77005
Karthik Kashinath NVIDIA Corporation, Santa Clara, CA 95051
Animashree Anandkumar California Institute of Technology, Pasadena, CA 91125; NVIDIA Corporation, Santa Clara, CA 95051



Conclusions PDEs

- Will ML play an important role in PDE solving?

- Important challenges:
  - Error guarantees→ trust
  - Data sparsity
  - Generalization
  - Stability
  - Multi-scale modeling
  - Non-regular grids
  - …



- Spectrum of methods: from traditional numerical solvers to completely data-driven surrogates
  - Where should we be on that spectrum?

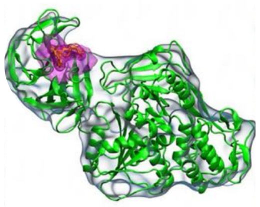
Molecules

Molecules

Everything material is made of molecules*

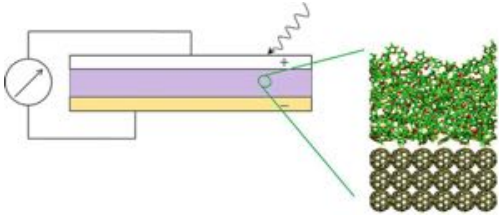
* Except 4 fundamental forces (electromagnetic force, gravity and strong & weak nuclear forces), and unless you break them up (plasma, quarks/leptons)

Molecules are at the root of solving many of the health, environmental and climate challenges we are facing today.



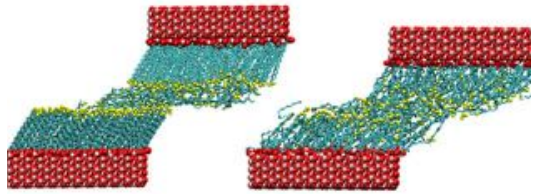
Drug discovery

Markus Reiher et al. PNAS 2017;114:29:7555-7560



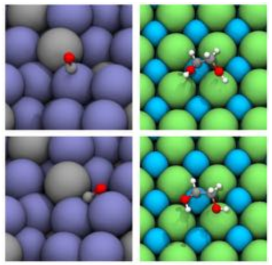
Photovoltaics

S.Y Reddy et al. Synthetic Metals 162, 23, 2012, 2117-2124



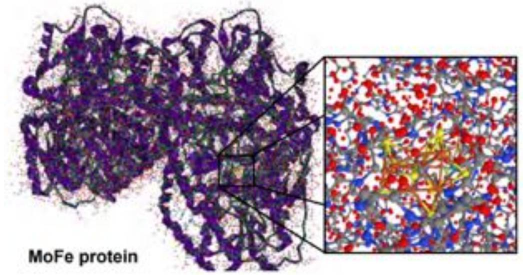
Tribology and lubricants

James Ewen, Tribology Group, Imperial College London



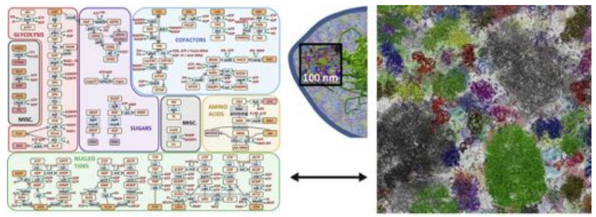
Catalyst design (e.g., fuel cells)

Lowik Chanussot et al. ACS Catal. 2021, 11, 10, 6059–6072



Nitrogen fixation

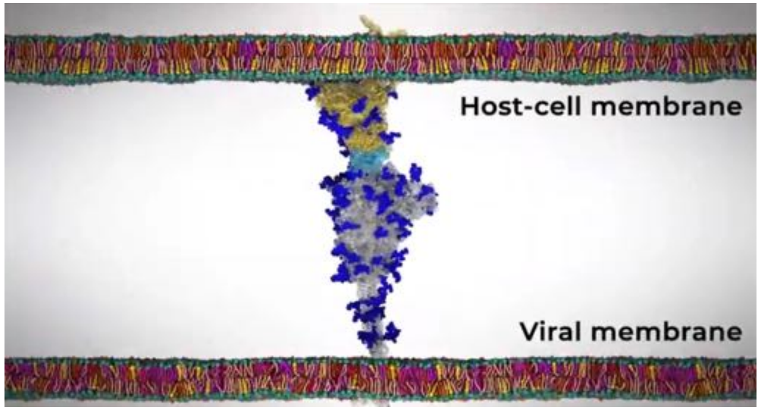
Shaher Bano Mirza et al. Journal of Molecular Graphics and Modelling 2016



Whole cell modelling

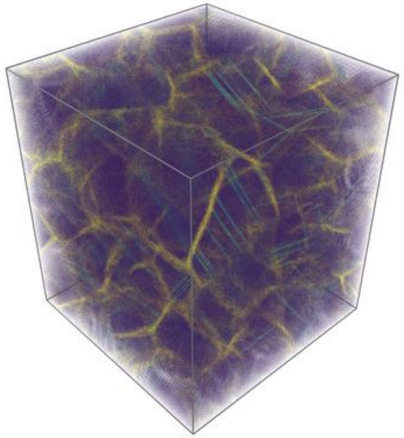
Michael Feig et al. Mol Graph Model. 2015 May ; 58: 1–9

Scale of Molecular Simulations is Huge



Lorenzo Casalino (UCSD) et al

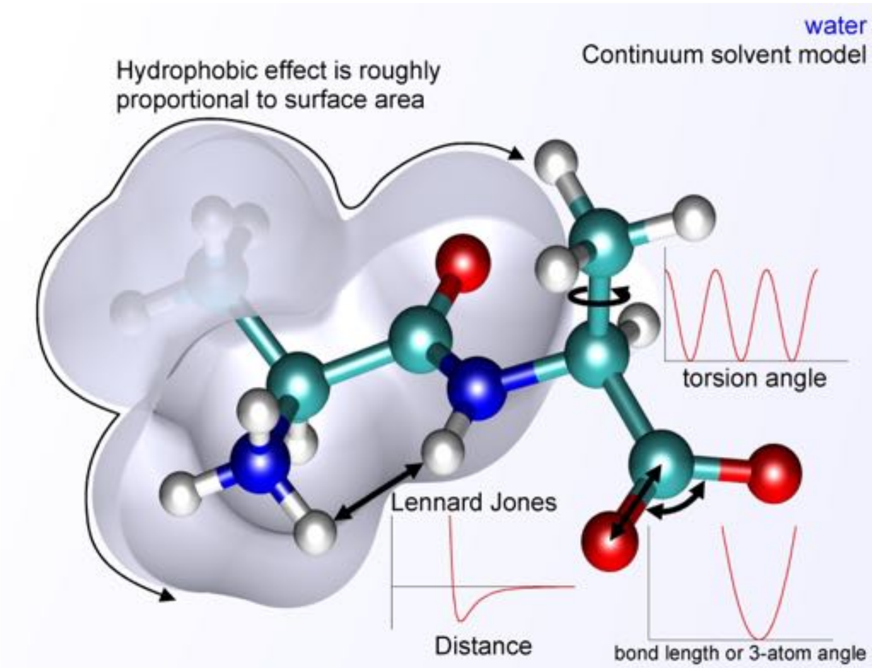
- Gordon Bell 2020 COVID-19 prize
- UCSD-led team of 35 researchers
- MD simulation of coronavirus
- 305M atoms
- 27,648 GPUs



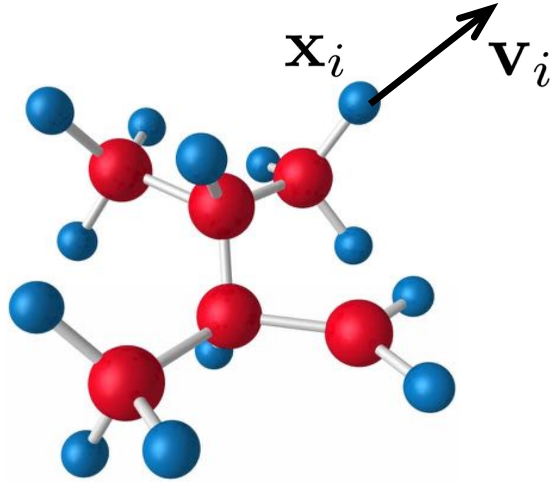
Weile Jia, et al

- Gordon Bell 2020 main prize
- Berkeley/Princeton/Peking collaboration
- MD simulation of metals
- 127M atoms
- 27,360 GPUs

Simulating molecules



$$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i$$
$$m_i \frac{d\mathbf{v}_i}{dt} = -\nabla_i U(\mathbf{x}_1, \ldots, \mathbf{x}_N)$$

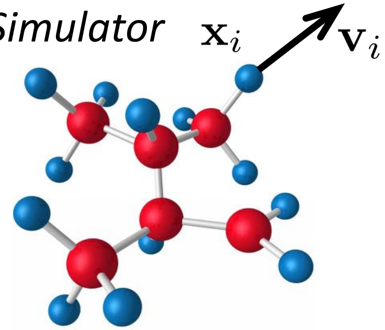


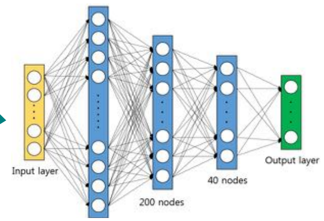
quantum mechanics

A *Search Engine* for Molecules

10^{180}	Upper estimate of the number of possible molecules
10^{80}	Estimated number of atoms in the observable universe
10^{60}	An estimate of the number of possible small organic molecules
10^{8}	The number of organic and inorganic substances in the CAS database

Simulator $\mathbf{x}_i$ $\mathbf{v}_i$

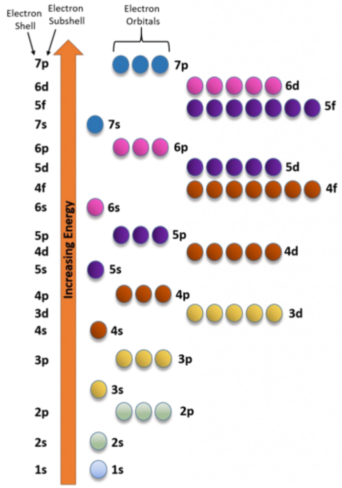




ML emulator

slow

fast

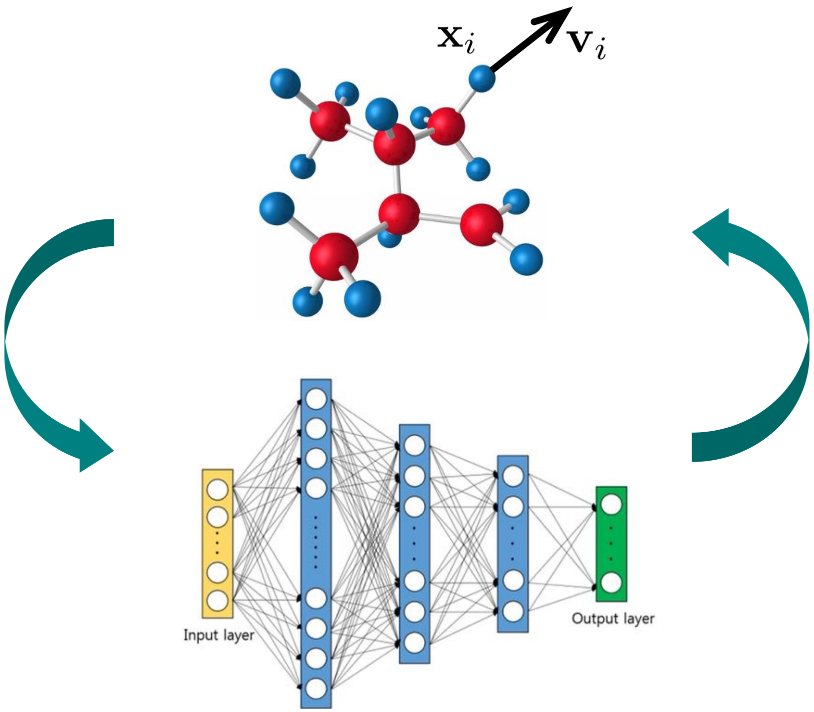


Molecular properties

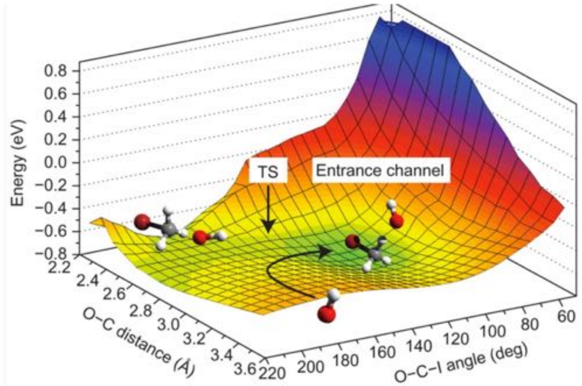
Inverse design: search space of molecules to find ones with prescribed properties

Some Examples: ML Forcefields

First principles *simulator*



Deep learning *emulator*

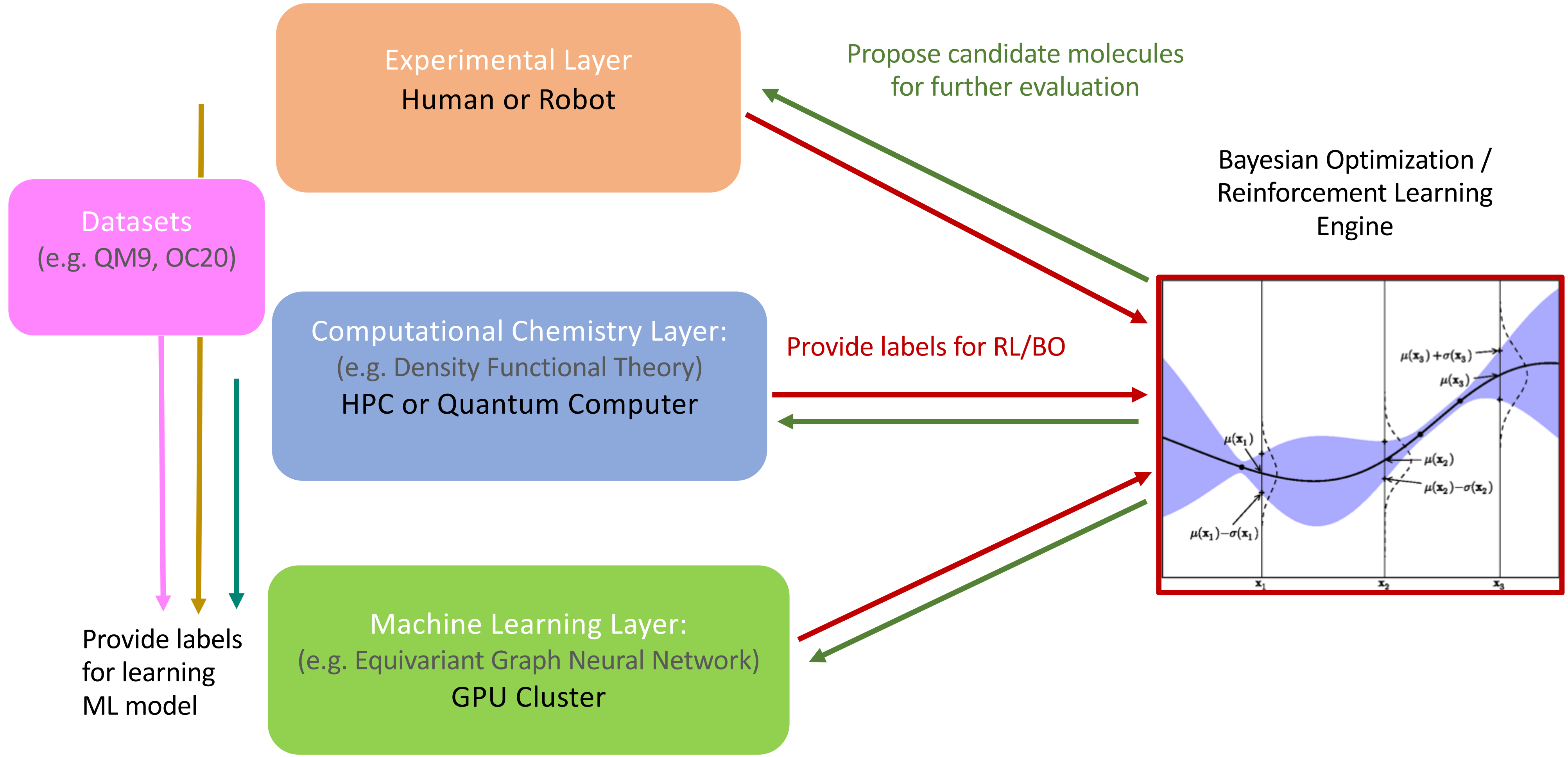


(R. Otto *et al.* 2011, *Nature Chemistry* 4, 534-538)

Synthetic training data
Perfectly labelled
Quantity limited only by compute
No privacy, GDPR, etc.

Data generation and **training** expensive
Amortized over many fast **predictions**

Reasoning over resources



Equivariant Normalizing Flows


E(n) Equivariant Graph Neural Networks

Victor Garcia Satorras[1] Emiel Hoogeboom[1] Max Welling[1]

Equivariant Diffusion for Molecule Generation in 3D

Emiel Hoogeboom[*1] Victor Garcia Satorras[*1] Clément Vignac[*2] Max Welling[1]

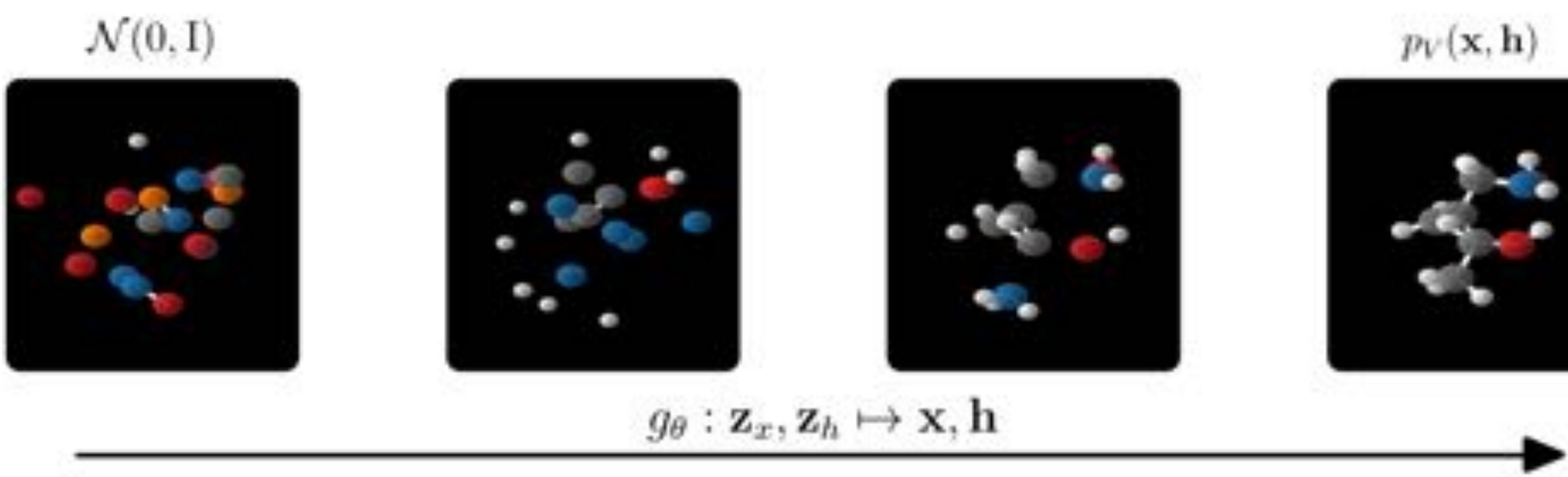
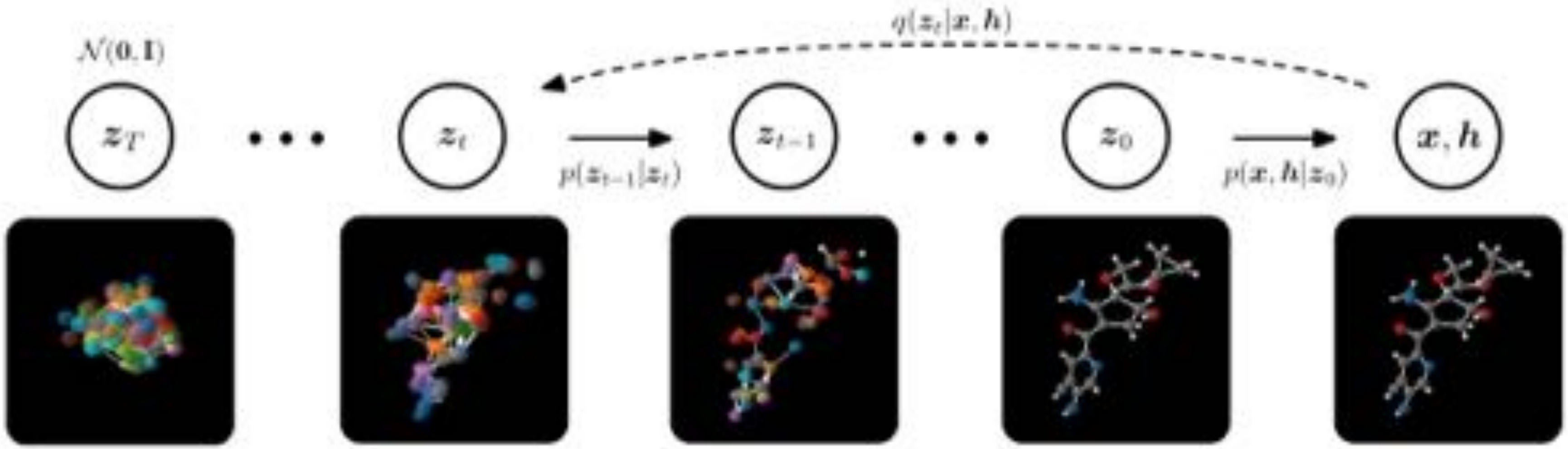
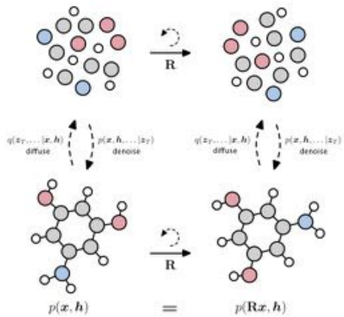


Figure 1: Overview of our method in the sampling direction. An equivariant invertible function g_θ has learned to map samples from a Gaussian distribution to molecules in 3D, described by $\mathbf{x}, \mathbf{h}$.

Diffusion Based Generative Models





Equivariant Diffusion for Molecule Generation in 3D

Emiel Hoogeboom *[1] Victor Garcia Satorras *[1] Clément Vignac *[2] Max Welling [1]

Molecule generation

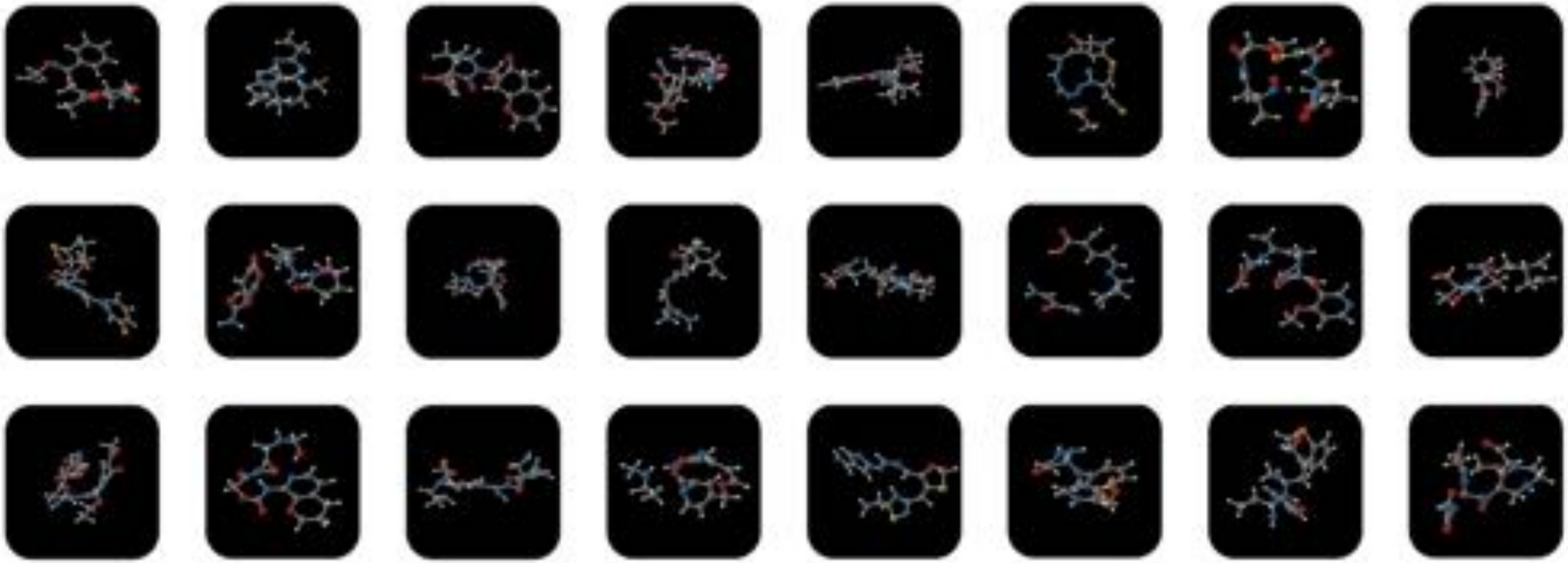


Figure 7. Random samples taken from the EDM trained on geom drugs.

Equivariant Diffusion for Molecule Generation in 3D

Emiel Hoogeboom [*1] Victor Garcia Satorras [*1] Clément Vignac [*2] Max Welling [1]

Holy Grail: Conditional (Equivariant) Generation

- Generate drug molecules with given properties (binds to disease, non-toxic, easy to synthesize)
- Generate material with prescribed properties (biodegradable, strong, binds to CO2, catalyzes a reaction)
- Accelerate MD simulation by generating proposal distributions

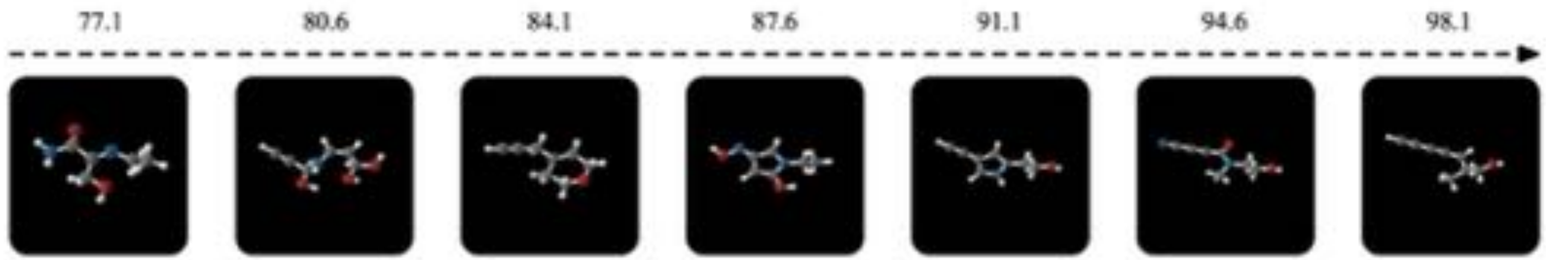
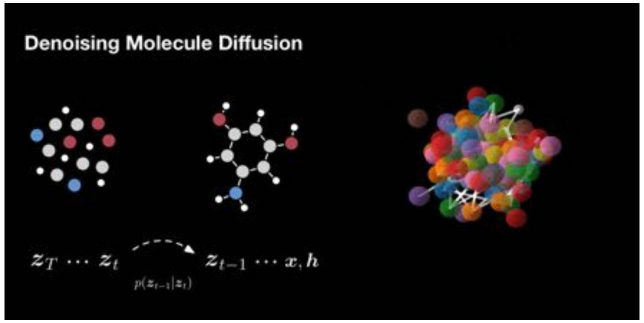
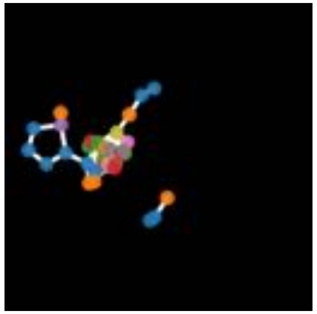


Figure 4. Generated molecules by our Conditional EDM when interpolating among different Polarizability α values with the same reparametrization noise ϵ. Each α value is provided on top of each image.

Generating Molecules and Materials





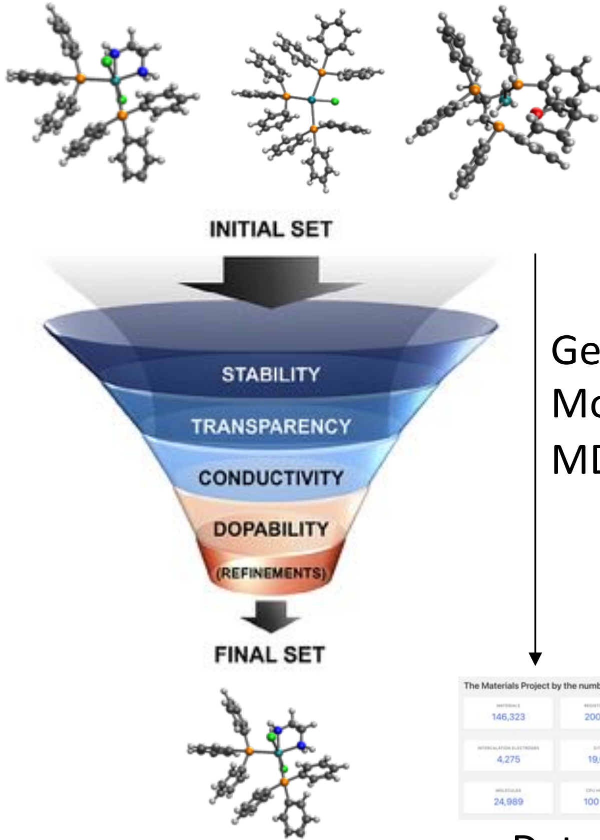
Equivariant Diffusion for Molecule Generation in 3D

Emiel Hoogeboom *[1] Victor Garcia Satorras *[1] Clément Vignac *[2] Max Welling [1]

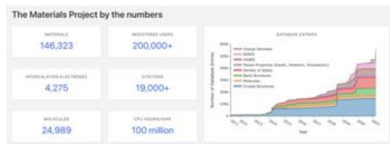
EQUIVARIANT 3D-CONDITIONAL DIFFUSION MODELS FOR MOLECULAR LINKER DESIGN

Ilia Igashov, EPFL, ilia.igashov@epfl.ch
Hannes Stärk, Massachusetts Institute of Technology, hstark@mit.edu
Clément Vignac, EPFL, clement.vignac@epfl.ch
Victor Garcia Satorras, Microsoft Research AI4Science, victorgar@microsoft.com
Pascal Frossard, EPFL, pascal.frossard@epfl.ch
Max Welling, Microsoft Research AI4Science, maxwelling@microsoft.com
Michael Bronstein, University of Oxford, michael.bronstein@cs.ox.ac.uk
Bruno Correia, EPFL, bruno.correia@epfl.ch

Molecule Generation
(e.g. for drug discovery)



Generative
Model +
MD finetuning



Materials Discovery

Data from e.g.
Materials Project

Quantum DFT Calculations

Mathematical foundations of DFT

Eric CANCES

Hamiltonian for multiparticle system:

$$\widehat{H}_N = -\sum_{i=1}^{N} \frac{1}{2}\nabla^2_{\mathbf{r}_i} - \sum_{i=1}^{N}\sum_{k=1}^{M} \frac{z_k}{|\mathbf{r}_i - \mathbf{R}_k|} + \sum_{1 \le i < j \le N} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} = \widehat{T} + \widehat{V}_{\mathrm{ne}} + \widehat{V}_{\mathrm{ee}}$$

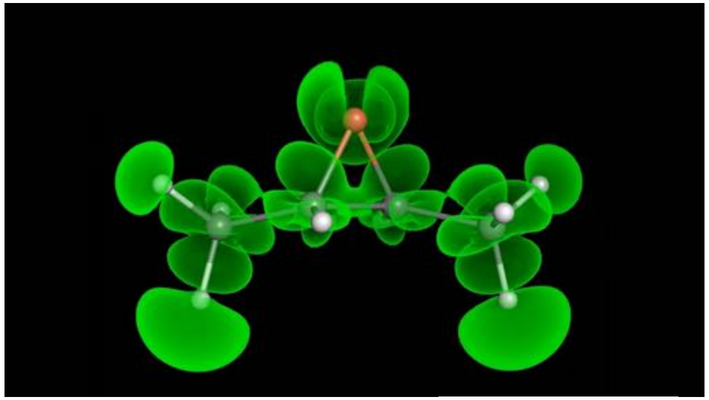
$$E_0 = \inf_{\Psi \in \mathcal{W}_N} \langle \Psi | \widehat{H}_N | \Psi \rangle$$

3n dim $\Rightarrow$ 3 dim

$$\boxed{E_0 = \inf_{n \in \mathcal{R}_N} \left(F_{\mathrm{LL}}[n] + \int_{\mathbb{R}^3} nV \right).}$$

$$\boxed{F_{\mathrm{LL}}[n] = \inf_{\Psi \in \mathcal{W}_N \,|\, n_\Psi = n} \langle \Psi | \widehat{T} + \widehat{V}_{\mathrm{ee}} | \Psi \rangle}$$

We don't know $F_{LL}(n)$ → learn from simulation data!



Quanta Magazine on DM21

Charlie Wood

Staff Writer

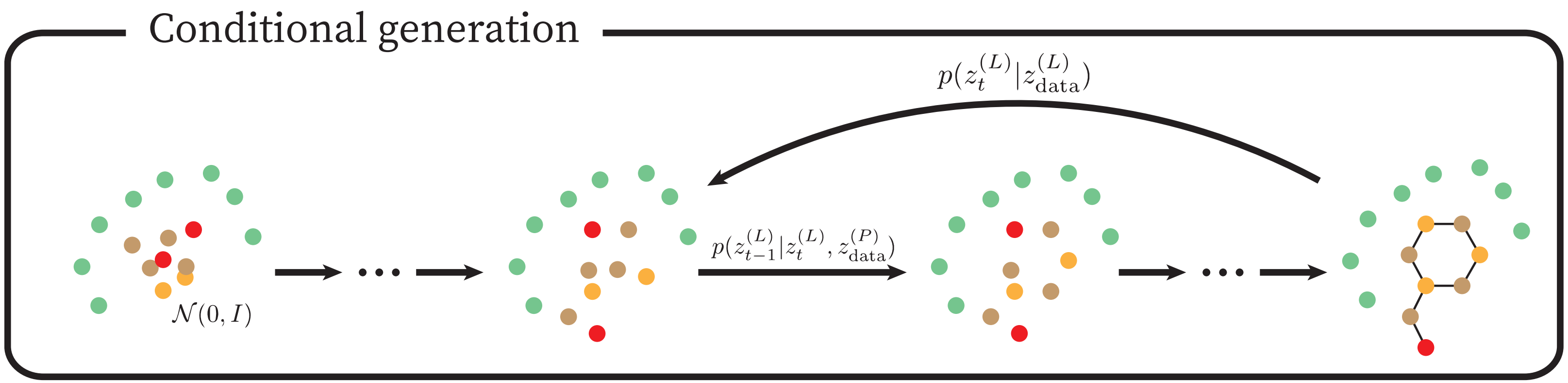
February 7, 2022

Approach: Conditional Equivariant Diffusion Model

Arne Schneuing

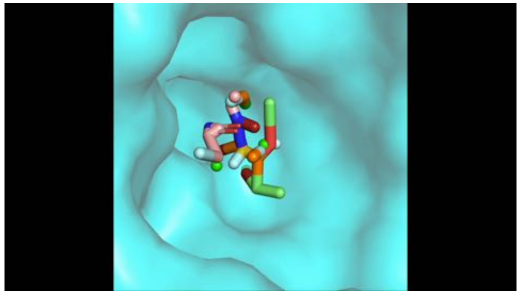
Ilia Igashov

(+B. Correira, M. Bronstein et al.)

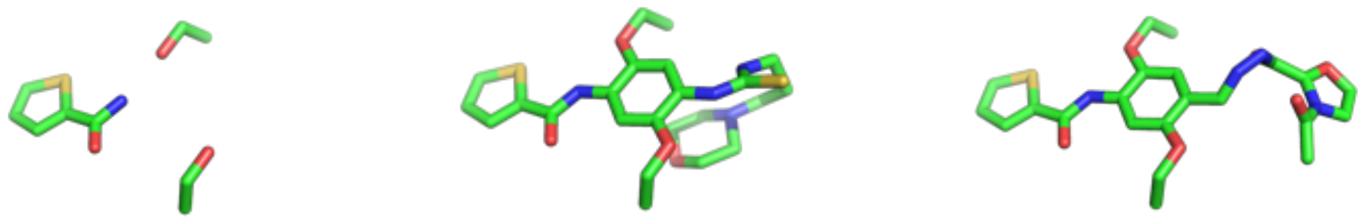


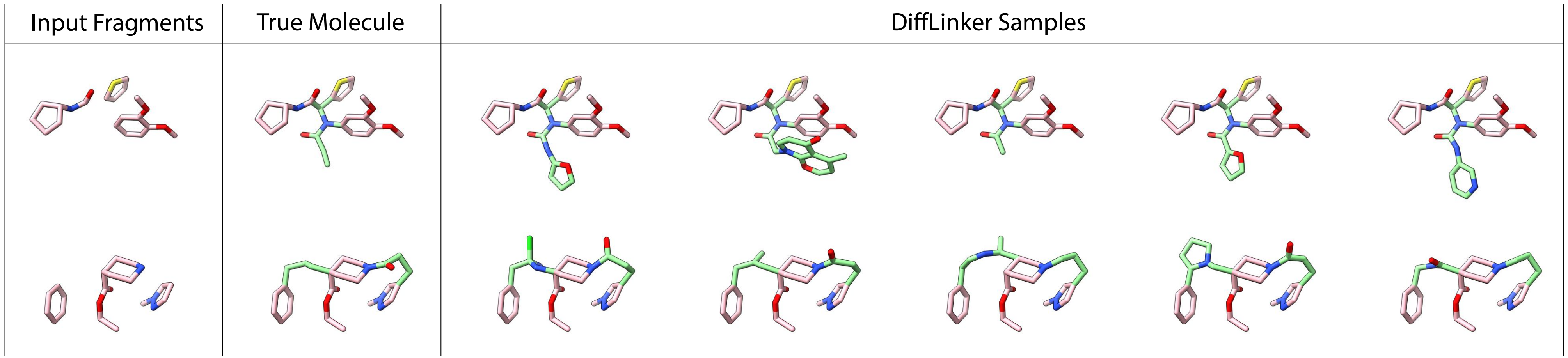
L denotes ligand nodes, *P* denotes pocket nodes

DiffLinker: Molecular Linker Design

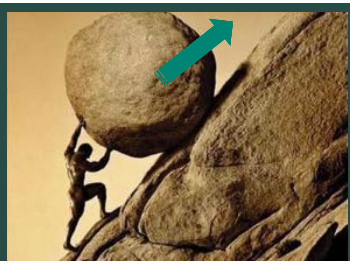


Input fragments Reference molecule Generated molecule





Transition Path sampling

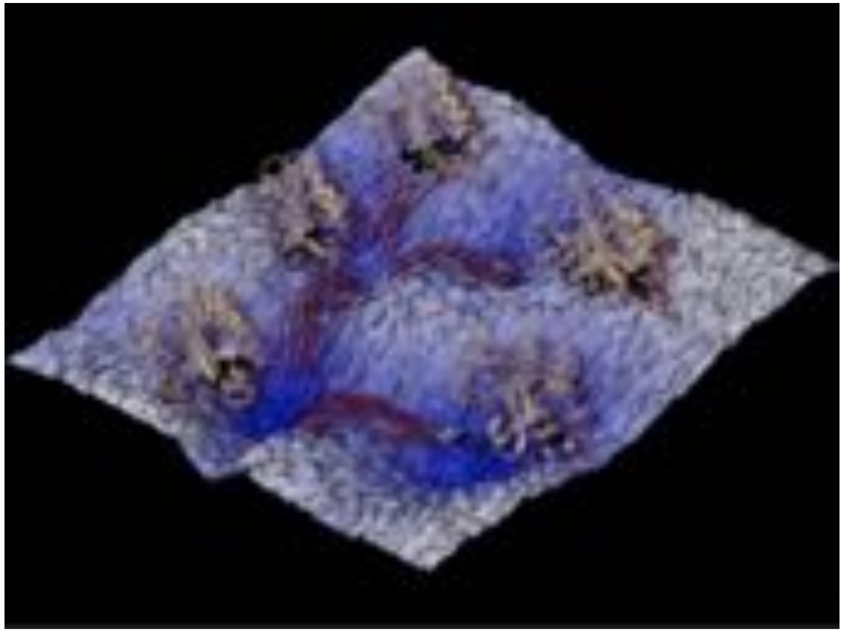


Project Sisyphus

Sampling transition paths between molecular conformations

PIPS : Path Integral Path Sampling

Given initial state r_0 and target state r_T find the series of intermediate states $\{r_1, r_2, \ldots, r_{T-1}\}$ that describe the transition path of minimal energy.



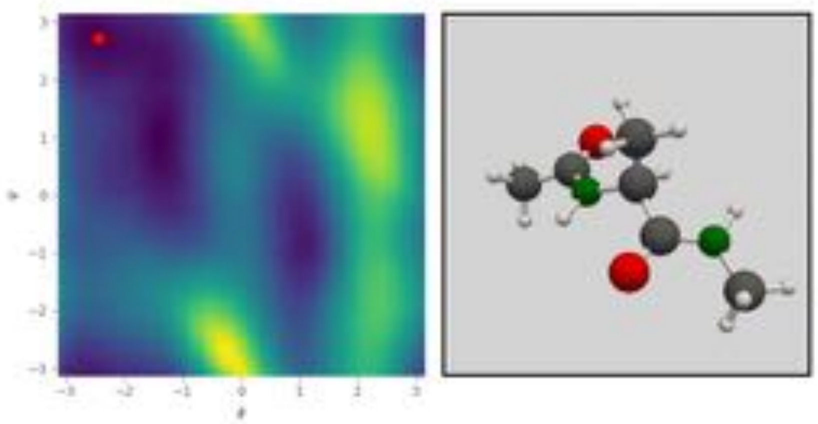
Source: https://www.e-cam2020.eu/rare-events-story/

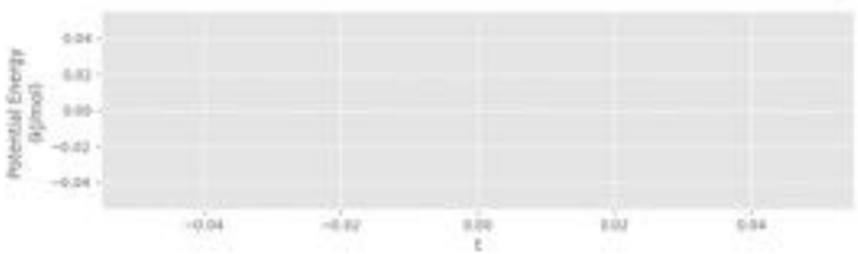
$$\underbrace{\begin{pmatrix} \mathrm{d}\boldsymbol{r}_t \\ \mathrm{d}\boldsymbol{v}_t \end{pmatrix}}_{\mathrm{d}x_t} = \underbrace{\begin{pmatrix} \boldsymbol{v}_t \\ -\nabla_{r_t}\boldsymbol{U}(\boldsymbol{r}_t) \end{pmatrix}}_{f(x_t,t)} \mathrm{d}t + \underbrace{\begin{pmatrix} 0_{3n} \\ \mathbb{I}_{3n} \end{pmatrix}}_{G(x_t,t)} \cdot \left(\boldsymbol{u}(\boldsymbol{x}_t, t)\,\mathrm{d}t + \mathrm{d}\boldsymbol{\varepsilon}_t\right), \qquad t \in [0, \tau]$$

Controlled dynamics

Alanine Dipeptide

- Extensively studied molecule with known collective variables





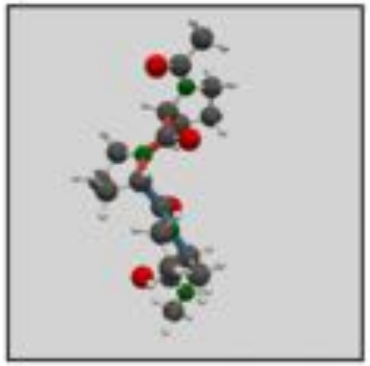
Collective Variables:

Dihedral angles ψ and ϕ

With Lars Holdijk, Yuanqi Du, Ferry Hooft, Priyank Jaini, Bernd Ensing

Polyproline Helix

- Transition from left-handed to right-handed helix (trans vs. cis)

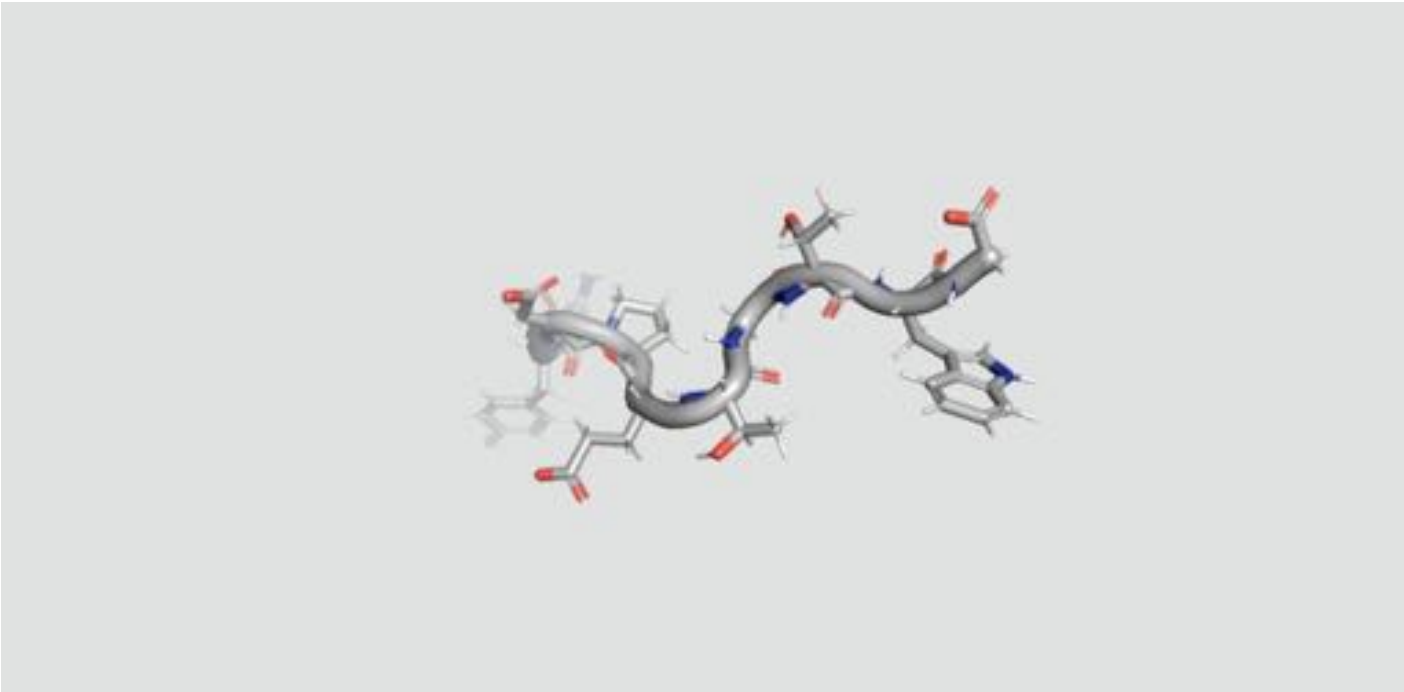


Collective Variables:

Trans vs. Cis helix

Chignolin

- Small artificial protein used for studying folding process



Collective Variables: Unknown

Molecules Represent a Huge Opportunity

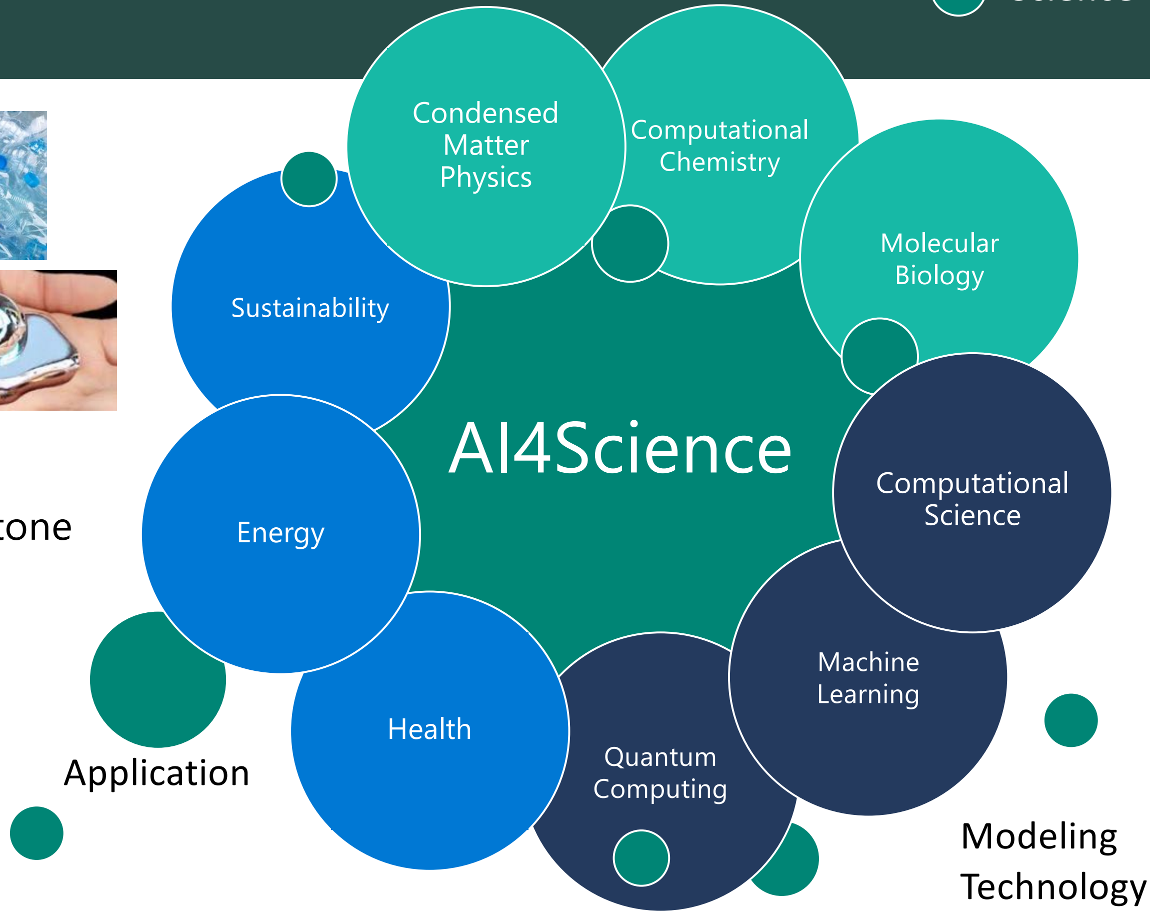




- We have named the ages of human development after the materials they use: stone age, bronze age, iron age, steel, plastic, aluminium,...*materials on demand?*

- A convergence of science, modelling technology and applications!

- A "golden age" of designing new materials/chemicals/catalysts/drugs?



Concluding Remarks

- Will ML change the way we will do science?
  Yes: building on the models in NLP and Computer Vision, ML will change the way we will do science.

- Huge opportunities to contribute to societal goals:
  - health (drug discovery, new antibiotics, vaccines)
  - Sustainability (carbon capture, battery materials, hydrogen production, synthetic fuels, nitrogen fixation,..)

- Huge economic opportunities:
  pharma, catalysis, materials